

TERRY KEPNER'S

portable 100

\$3.95/CAN \$4.95



THE DOG ATE MY HOMEWORK!

Your computer can teach anything to anybody! 2 Free programs inside.

A 3-D profile of your students with the *Grade Management System*.

LapDOS II:
The PC meets
the portable
disk drive!
And the
notebook
computer!

HOTKEY:
Extra features
instantly
at your
fingertips.



0 186036 8

Announcing:

The SOUNDSIGHT GOLD CARD

Megabytes of memory on a credit-card-sized wafer
A mere 1/8" thick

Give your **Tandy 100/102** the power it deserves without sacrificing portability or adding bulky boxes. The SoundSight Gold Card lets you add as many megabytes of **RAM and ROM** as you wish with advantages other expansions just can't match.

- Allows **Unlimited RAM and ROM** expansion in increments from 128k to 1mb per card with 2 mb cards available in '88. Additional slots piggyback allowing 10 mb or more!
- **Dual Card Slots** with dual directories allow easy transfer of megabytes of data from card to card or computer to computer in seconds.



• Replaceable,
Card-Contained
Lithium Battery.

• **Menu Driven ROM** resident operating system with Basic I/O functions, read/write error detection, and nondestructive RAM test. Accesses Text, Basic and Co files directly and operates in Random Access Mode. Each byte is directly accessible from Basic.

• SoundSight Gold Cards are also available in permanent **One-Time-Programmable** or **Read-Only-Reprogrammable** memory cards for large nonvolatile software application use in conjunction with RAM cards.

Optional new **ROM ELIMINATOR** lets you load and run multiple ROMs directly from the SoundSight Gold Card, dispensing with the need for extra ROM sockets or even having your ROM chips with the computer!

NEW OPTIONAL SOFTWARE DEVELOPMENTS:

- ROM resident **Forms Generator** with full Telecommunications and Desk Top package for Law Enforcement Incident Reports, or any custom data entry applications.
- Point of Sale **Inventory Control Software**
- New **Basic Compiler** compiles basic programs to machine code, condensing them to fraction of their original size and speed. Runs them directly from cards. (Avail. Feb. '88.)
- New **Text Editor** opens and edits files as large as available ram. (Avail. Feb. '88.)



SoundSight offers **customized hardware/software** systems for law enforcement, retail, warehousing/distribution and other vertical markets, as well as third party dev. (C language).
We're ready to meet your needs TODAY!

Pricing available from 128k to 10mb with 128k at \$399.95; 256k at \$549.95; and 512k at \$749.95; Additional cards for backup are available.

CONTACT:

SoundSight MBM Inc., 225 West Broadway, Suite #509, Glendale, CA 91204
Phone (213) 463-9457 or (818) 240-8400

Circle 22 on reader service card.

FINGERS TIRED?

Accomplish more... in less time.

Get the P100-To-Go monthly program disk.

Make your life easier! Save hours of time from typing in lengthy programs. Never worry again about typing errors. Have more fun with Portable 100's new disk-of-the-month service. Every program from the magazine is included on each month's disk. No typing. No trouble.

We make it easy and inexpensive. You deserve value. At Portable 100 we believe in value. That's why we're not only making each month's programs available on 100K Tandy Portable Disk Drive disks, we're also making sure any extra room on each disk is packed full of interesting and fun bonus programs. All files are in .DO format. Ready to load. Ready to go.

Get more from your Notebook computer. Organize your life and be more productive with **home and office management** programs. Expand your knowledge with **educational** programs. Relax and enjoy

games or simulations. Make using your Tandy a snap with helpful and handy utilities. Easy to use. Always available.

Subscribe now and save. You can order Portable 100's *P100-To-Go* disks individually at only \$9.95, but for maximum savings sign up for our **Six-pack Special**, six disks, six months of programs for only \$47.70. Save over 30%, a savings of \$2.00 per disk.

Not just for the Tandy 100. The *P100-To-Go* disks will work with any computer that uses the Tandy Portable Disk Drive, the Brother FB100, and the Purple Computing Portable Disk Drive. This means your NEC-8201/8300, Tandy 100/102, Tandy 200, Olivetti M10, and Kyotronic KC-85 can all load these programs into memory (once loaded, changes for specific computer models may be required, refer to corresponding program article for information about compatibility with particular computers).

- ☐ **Yes!** Save me hours of typing time. Sign me up for your **Six-pack Special** (6 monthly disks of the Portable 100 programs, plus the special bonus programs included with every disk) for only \$47.70.
- ☐ June '88 single issue *P100-To-Go* disk, \$9.95.
- ☐ July/August '88 single issue *P100-To-Go* disk, \$9.95.
- ☐ September '88 single issue *P100-To-Go* disk, \$9.95.

Prices include postage and handling. Please allow 4-6 weeks for delivery. Foreign airmail add \$2 per disk. All foreign, US Funds drawn on US banks only.

- ☐ Payment enclosed (check or money order)
- ☐ Mastercard ☐ VISA ☐ Amex

To place your order immediately, call: 1-603-924-7949

CARD NUMBER

EXP. DATE

SIGNATURE

NAME

STREET ADDRESS

P.O. BOX

CITY

STATE ZIP

Mail to: Portable 100, P100-To-Go,
P.O. Box 428, Peterborough, NH 03458-0428

ON
THE
COVER:
Toni Gildone's
dog Everett
munches out on
a Tandy 102.

Photo by
Mike Nugent and
Dave Klein.



VOL. 5, NO. 8
SEPTEMBER 1988



Tandy 102

TEACH YOUR TANDY TO TEACH

by David Dunn Thomas

A Model 100/102 tutoring program for the teacher and student.

6

A NOTEBOOK-COMPUTER TEACHER

by E.A. Schwartz

Tutoring and testing with the Tandy 100/102, 200, Olivetti M10, and NEC 8201.

10



Tandy 200

A DANDY TEACHER'S AID

by David Klein

A thorough review of Software by Steffens' Grade Management Program.

14

IBM AND THE TANDY PORTABLE DISK DRIVE

by Mike Nugent

LAPDOS II lets you use your Tandy Portable Disk Drive with both your Tandy/NEC notebook computer and your IBM PC or clone.

16



Tandy 600

ENDING YOUR COMPUTER'S SOLITUDE

by David O. Rowell

This month make a null modem cable (which lets two computers talk directly to each other) next month we'll have a practical application requiring such a cable.

19

HOTKEY: ESCAPE ARTIST

by Mike Nugent

Keyboard macros (and then some) for the Tandy 100/102.

21

UTILITYCORNER

DOS PATCHES

by Mike Nugent

FLOPPY.CO patches to prevent clashes with low-memory machine-language programs.

22

DEPARTMENTS

ROM WITH A VIEW

3

I/O

4

DEFUSR

25

THE CUSTOM 200

27

ADVERTISER'S INDEX

PORTABLE 100 CLASSIFIEDS

32

ROM WITH A VIEW

Back Issues! Stashed away under bales of hay in Camden, Maine, we have found the coveted *Portable 100* back issues! Unfortunately, we have only about a hundred copies of each issue and we had to pay through the nose for them. Reprints of articles are available, along with a comprehensive article index. The issues themselves are *collector's items* and are being sold for \$19.95 each, on a first-come-first-serve basis. See page 30 details.

PRINTING MONEY is not the only way to make your Model 100 generate cash. We get a lot of letters from readers about the lack of products available. It is frustrating to have one of the most powerful and convenient computers available and have nothing to do with it.

For those of you who are patiently awaiting an invitation to put the programs you've written on the market...consider yourself invited! Now is the time to turn your hard work into cash. Most of the big players in the industry started in their garage or cellar, hacking away at their computers. Look at Apple or Microsoft. Another example is *Grade Management System* by Software by Steffens (reviewed in this issue). Gary Steffens is a teacher with a good idea now selling his program. It just takes some skill and guts.

Here's how it works:

- 1) Take the best program you've written, spruce it up a bit, and write some non-techie documentation. Get someone who is not as good at computers as you and have him/her test your program for bugs and other problems. If possible, try it on different hardware (tape drive, disk drive, etc.).
- 2) Take out a classified ad in three consecutive issues. Some of the best deals out there are available through classifieds and our readers know that. We'll help out if we can, reviewing appropriate products in the magazine.
- 3) Use the money from the first few month's sales to improve the program and manual, and take out a small display ad. Keep an effective mailing list and make sure that the people who bought the first copies get upgrades.
- 4) Relax a while and rake in the money. Although running a business takes hard work, there is no reason why you can't earn extra money and give yourself the opportunity to realize some of your dreams.

There are hundreds of thousands of computers out there. Many are looking for new, practical, and innovative ways to use their computers. If you have the ability to offer solutions, you can earn your "piece of the pie."

- Mark Robinson

Toolbox

Manuscripts were typed into Microsoft Word 3.0 on a Tandy 1400 LT (dual drive, 640K), where they were edited, spell-checked, and had basic format instructions inserted. From there they were loaded into a Tandy 4000 (80386 CPU, Tandy EGA Monitor, Tandy LP-1000 LaserPrinter) desktop computer and placed into Aldus' IBM PageMaker 2.0a. There they were dummied into a rough approximation of the magazine's final appearance. Here, pull quotes are placed, headlines, intros, and bylines are sized and positioned, and advertisements are allocated space.

Next, the magazine (divided into sections) was ported over to Diana Wallace's Macintosh Plus, using the

1400 LT and Mac-link. Diana then went over the publication using Aldus Macintosh PageMaker 2.0a, page-by-page, making final design decisions on photo, figure, and listing sizes and placements. She precisely placed the text and added all the little things that go into making a nice looking publication.

Approximate page previews were output from her Apple LaserWriter Plus. When everyone was satisfied with the appearance, the Macintosh disk was sent to Colorite Corp. in Wisconsin for final output directly onto film (and in some cases, photographic paper). The film was then delivered to the printer, who printed it, labeled it, and mailed it to you.

CHIEF EXECUTIVE OFFICER
Marlene Butland

PRESIDENT/PUBLISHER
Terry Kepner

EXECUTIVE VICE PRESIDENT
Mark Robinson

EDITOR-IN-CHIEF
Terry Kepner

FEATURES EDITOR
David Klein

TECHNICAL EDITOR
Mike Nugent

NEW PRODUCTS EDITOR
Linda Tiernan

CIRCULATION MANAGER
Shauna Crowley

ADVERTISING DIRECTOR
Randy Byers

**ART DIRECTION and
DESKTOP PUBLISHING**

Diana Shonk
ACROSS THE BOARD
Graphic Design, Inc.
8 Windy Row
West Peterborough, NH 03468-0091

**PORTABLE COMPUTING
INTERNATIONAL
CORPORATION**

145 Grove St. Ext., #21, PO Box 428
Peterborough, NH 03458-0428

Editorial
603-924-7949

Advertising
603-924-9455

Circulation
603-924-9455

Portable 100 (ISSN 0893-942X) is published by Portable Computing International Corporation, 145 Grove Street Ext., P.O. Box 428, Peterborough, NH 03458-0428. *Portable 100* is an independent journal not connected with any hardware, software, or peripheral equipment manufacturer. *Portable 100* is published monthly, except for a combined July/August issue in the summer. Entire contents Copyrighted 1988 by Portable Computing International Corporation, All Rights Reserved. No part of this publication may be reproduced without written permission from the publisher. Portable Computing International Corporation makes every effort to assure the accuracy of articles published in *Portable 100*, but assumes no responsibility for damages due to errors or omissions. Subscription Service: All subscription correspondence should be addressed to *Portable 100*, Portable Computing International Corporation, 145 Grove Street Ext. P.O. Box 428, Peterborough, NH 03458-0428. U.S. subscription rates: \$24.97, one year; \$53 two years. Canada and Mexico: US\$29.97, one year; US\$61 two years. All other foreign (surface mail): US\$44.97, one year; US\$85 two years. Foreign Air Mail, add US\$50 per subscription year. All payment, except Canada, U.S. funds drawn on U.S. Bank. Second-class postage paid at Peterborough, NH 03458, and at additional mailing offices.

POSTMASTER: Send address changes to: *Portable 100*, Portable Computing International Corporation, 145 Grove Street Ext., P.O. Box 428, Peterborough, NH 03458-0428.

A Better Power Pipe

In the March '88 issue Kevin Martin describes how he built a "five dollar" Power Pipe for his Model 100. Mine came from the local Tandy store and cost nothing at all (see photos below)!

Obtain a Tandy 5-cell torch [*flash-light*] catalogue number 68-1001. Over here they are often given away or sold very cheaply in connection with Tandy's famous sales. Cut off the business end above the switch and fix the legs and wires as described in Kevin's article (the torch has a metal strip down the inside for the negative connection) and there it is. I use mine to power both the computer and disk drive, and the added bonus of the switch means there is no live lead flopping about when the unit is disconnected.

I'm not sure what "telescope tubing" is. For the positive connection I used the spring ring from a coaxial TV socket which was just the right size. Thanks for all your efforts with the magazine, they are appreciated.

John Bloxham
Stratford-upon-Avon
Warwickshire, England

NOT ENOUGH COVERAGE

I own a Tandy 600, and unfortunately you do not run enough articles

on this machine to justify my continuing the subscription. You do a fine job for the Model 100/200 owners, and were you to increase coverage to the 600, your magazine would be much more useful to me. I will continue to buy your magazine from the rack when there are articles dealing with the 600 or info I can use.

Kenneth F. Taylor
Portland, OR

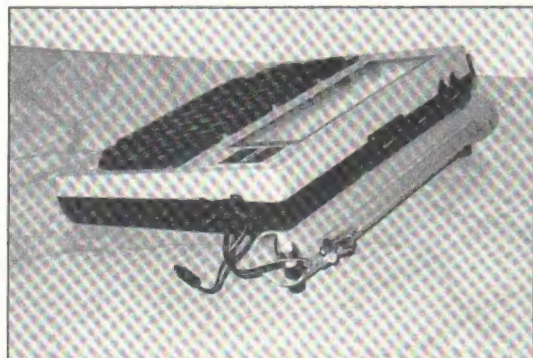
Mr. Taylor, please read on....

*We feel positive
about this joining
of resources.*

I am a professional programmer who is very interested in the Tandy 600 laptop and very concerned about the lack of software and information on the machine. I not only develop software, but I am currently writing several articles about the 600. I have

been able to add disk drives to the 600 while making the disk format compatible with the PC-DOS and MS-DOS computers. Enclosed is the first edition of our Club 600 newsletter, which we are using as a tool to explain the Tandy 600—and some of our products—to Tandy 600 owners. We will be covering a huge amount of material in a broad and not-too-technical sense. We will be working hard to find new and exciting information, tips and tricks, and of course, we will be announcing new products in the near future. Feel free to print any or all of this material, so long as myself, CLUB 600, or Easy Ware Products are mentioned. This laptop has a massive amount of potential; I have not yet seen a rival which comes close to its price and capabilities. And we receive letter after letter from people who feel the same way. Thank you for the support and offerings. We feel positive about this joining of resources and hope Tandy 600 users will benefit most from it. Let's make this the year of the laptop, and especially the Tandy 600.

Dennis C. Rogers, President
Easy Ware Products
P.O. Box 218778
Houston, TX 77218-8778
Tel. 713-933-0542



I am a recent subscriber to Portable 100 magazine as well as a relatively new owner of a 200 with a PDD2. I enjoy and look forward to your magazine each month, but I have one complaint. It is too short!

I, too, would like to see more articles for

the 200 in upcoming issues. Things like product reviews and comparisons of software would be greatly appreciated. The products for portables seem to be awfully expensive, especially the software. No one feels comfortable paying a couple of hundred dollars for a program they know next to nothing about.

Also, how about offering a disk or cassette with the programs featured in your magazine? This would save a lot of headaches for us hunt-and-peck typists.

Kyle B. Pauley
New York, NY

MS. LEGEZA: A SOLUTION

Naturally, I can relate to letters like the one from Louise Legeza and Dr. Wilson's lengthy reply. I solved Louise's problem with David Lien's book, a learner's manual for the Model 100. I swear that man could teach a fish to fly and a bird to swim. Come to think of it, maybe he's the one that did!

Eric N. Isakson
Neenah, WI

BACK ISSUES ARE HERE!

Portable 100 is a refreshing publication among the hordes of computer magazines. I enjoy reading about what's being done with the Model 100.

Thanks for publishing the article by Hal Boulware on re-inking DMP-105 ribbons. That article alone has probably justified the cost of subscribing to *Portable 100*.

Is it possible to get back issues of *Portable 100*, back even to Sept. 83?

The review of "Booster Pak" was pleasingly thorough—I'd like to read a review of "Super ROM" and would be fascinated to read about the people behind the scenes at PCSCG.

I also would be interested in an article about the SoundSight Gold Card. Does it really do all the stuff they promise in their advertisement?

John S. Neufeldt
Tucson, AZ

LETTER FROM THE EDITOR:

M100 vs. T102 vs. T200 vs. T600 vs. T1400LT vs. PC8201 vs. PC8300 vs. KC85 vs. M10 vs. BASIC vs. Assembler vs. Beginner vs. Advanced vs. Business

vs. Games vs. Software vs. Hardware....
All-of-the-Above vs. 32 pages!

Quite a challenge, but unless we pull it off, you're pretty much left without a source of information and support. So we have to find ways to make it work. Club 600 now provides us with a wealth of Tandy 600 goodies. The LAPTOPS Roundtable on General Electric's GENie system has great NEC and Tandy 600 support in addition to the 100/102/200 fare. CompuServe has lots to offer as well. We'll be covering more of Tandy's 1400LT now, too.

Where possible, I'll be testing programs in the magazine for compatibility on several machines. So don't just skip that Model 100 article—it probably works on your NEC 8201 with a few indicated changes. (LEARN.BA in this issue works on almost everything.)

Pre-testing programs will also help cut down on typos and bugs in the programs we publish. And yes, Mr. Pauley,

*I'm demented,
and
I never sleep.*

programs will be available on disk; look for an ad elsewhere in this issue. They'll soon be on our PICO BBS as well, along with the FORTH programming package for Mr. Iverson. (See "DEFUSR.")

Mr. Neufeldt can get back issues (which have reviews of Super ROM and others) through the advertisement on page 31 of this issue. And he can look forward to a review of the Gold Card and other SoundSight products. It's installed on my Model 100 right now. Can anybody help him with his ribbon re-inking problem? (See "DEFUSR.")

We can't neglect one segment of our readership in favor of another. If we focus on beginner's tutorials, where will the beginners go when they're no longer beginners? We're going to cover it all as best we can.

It's quite a challenge, but I'm de-

mented, and I never sleep. We'll pull it off!

Mike Nugent
Editor in Charge of
Working Real Hard,
Ministry of Silly Walks

THE MYSTERY OF THE FLAT SQUIGGLE

You did not answer Mr. Wright's inquiry (June '88, p. 7) about the strangeness of the equation on p. 19 of the article "The Missing Link" (Feb. '88). Once more, and this time it's me asking the question, what is the following?

(P/X)~E

I have tried to get the answers given in the article by assuming that this was meant to be a scientific notation, but that doesn't seem to be the answer either.

You might also note that I have this "flat squiggle" character on my Olivetti M10, but that it does not even exist on the Tandy Model 102. So, what is it?

Will you please finally answer the question before we all loose [sic] our minds? By the way, I do have training in advanced mathematics, unlike Mr. Wright, and I still can't figure it out! Editorial precision triumphs again, and again, and....

Norman L. Donaldson
Granada Hills, CA

That flat squiggle (~) is a tilde, CHR\$(126) on the Tandy 100/102/200, NEC 8201, and Kyotronic KC-85. On all but the NEC, you can produce it directly by pressing the GRPH-[] combination. The character which should have appeared in the article is the caret (^), CHR\$(94), found on the top row of the Tandy 102 keyboard, and is intended as an exponentiation operator. As much as possible, I'll be testing programs before publication, so these errors should become much less frequent in the future.

MN

We welcome all letters from our readers, whether critical or complimentary. We print as many letters as space permits (some are edited for space considerations). Address your correspondence to: *Portable 100*, I/O Dept., P.O. Box 428, Peterborough, NH 03458-0428.

COMPATIBILITY: Tandy 100/102 only

★ Teach Your Tandy To Teach ★

This powerful BASIC program never criticizes your mistakes.

by David Dunn Thomas

Scholar Reviews provides teachers a medium for exposing students to new, or reviewing old, material. The program uses lessons on file—in RAM, a Tandy portable disk drive, or a Chipmunk drive—to provide the questions, background information, and the correct answers. The program is only 3,274 bytes in BASIC format. Each lesson may have up to one hundred questions, depending on the amount of review data provided. You can prepare dozens of lessons, stored on one disk or in one folder of a Chipmunk disk, for one study session.

The important feature of *Scholar Reviews* is the built-in review capability of the program. Whenever the student selects an incorrect answer, the program promptly provides information, including the correct answer to the question. When a student answers a question correctly, it repeats the answer, enhancing learning. Also, the teacher may direct the program to repeat an answer but phrase it differently from that of the question.

The *Scholar Reviews* program file, *STUDY.BA*, is simplicity itself for the student to operate for learning or reviewing a lesson. On the other hand, the more difficult and painstaking operation is the preparation of the lesson files so that the *Scholar* can make proper use of *STUDY.BA*.

USING STUDY.BA

When a student starts running *STUDY.BA*, it asks for the lesson to review and whether the lesson file is in RAM or on disk. If the file does not exist, the program drops the user out of the program with an appropriate advisory message. If the file does exist, but is not in the proper *Scholar Reviews* format, it drops the student out of the program when it comes to the incorrect formatting in the question/answer phase of the program.

Once embarked, the student has a series of multiple choice questions with either four-choice or true/false questions. The program does not chastise the student for

```

0 ' STUDY.BA v2.2 "The Scholar Reviews"
1 ' Copyright 1988 by David Dunn Thomas
2 ' All rights reserved
7 GOTO14
8 DATA 32,84,104,101,32,83,99,104,111,10
8,97,114,32,82,101,118,105,101,119,115,3
2,97,114,101,32,97,32,115,104,97,114,101
,119,97,114,101,32,32,32,32,112,114,1
11,100,117,99,116,32,99,111,112,121,114,
105,103,104,116,32,98,121,32,68,97,118,1
05,100
9 DATA 32,68,117,110,110,32,84,104,111,1
09,97,115,32,241,241,241,241,241,241,241
,241,241,241,241,241,241,241,241,241,241
,241,241,241,241,241,241,241,241,241,241
,241,241,241,32,83,101,110,100,32,36,36,
32,116,111
10 DATA 58,32,50,51,48,56,32,67,104,101,
116,119,111,111,100,32,67,105,114,99,108
,101,44,32,49,48,51,32,32,32,32,32,32,32
,32,32,32,32,32,32,32,32,32,84,105,109,111,
110,105,117,109,44,32,77,68,32,50,49,48,
57,51
11 FORJ=1TO8:CALL17909:NEXT:FORIP=1TOLEN
(P$):POKE64984+IP,ASC(MID$(P$,IP,1)):NEX
T:FORJ=1TOLEN(PV$):POKE64783+J,ASC(MID$(
PV$,J,1)):NEXT:RETURN
12 CLS:PRINT@55,R$ " Correct! "N$:IFCF THE
NRETURNELSEC=C+1:CF=0:RETURN
13 CALL21795:CALL16974:CF=-1:RETURN
14 CLEAR2000:DEFINT A-Z:DIMP(320):E$=CHR$(
27):R$=E$+"p":N$=E$+"q":Z$=CHR$(124):ZT
$=CHR$(126):P$=SPACE$(23)+"Press any key
->":CLS:PRINT@89,R$ " T H E S C H O L A
R "N$:PRINT@134,"v2.2 6/3/87":PRINT@20
7,CHR$(171)" 1988 by David Dunn Thomas"
15 RESTORE:ONERRORGOTO46:FORI=81TO271:RE
ADP(I):NEXT:FORT=1TO2500:NEXT

```

Continued.

A program to help you review information or tutor others.

being wrong; rather, it provides the student with the correct information and then it asks the question again. When the student answers the question correctly, the answer is returned to the student—possibly rephrased. If the student wants to leave the program at any time, pressing the ESC key at any question prompt jumps the program to an ending routine (line 43) and exits the program. For example:

A president of U.S. during World War I was:

- 1) Woodrow Wilson
- 2) Jack Pershing
- 3) John Rockefeller
- 4) Benjamin Franklin
- ?

If the student chooses 3), an incorrect response, the screen displays this:

Pershing, Franklin, and Rockefeller were never presidents of the U.S., but Wilson served from 1913-1921.

★ Lesson formatting
provides for
instructing the student
instead of merely drilling—
one of the principal
strengths of the
Scholar Reviews concept. ★

With the correct response, the display is:

*Correct!
Woodrow Wilson
was president in WWI, 1914-1918.*

Note that lesson formatting provides for *instructing* the student instead of merely *drilling*—one of the principal strengths of the *Scholar Reviews* concept.

For some idea of the capabilities of *Scholar Reviews*, run the program *STUDY* (see Listing 1) with the *LESSON.DO* file (see Listing 2) loaded into either RAM or disk. There are only sixteen questions in the file, but these are sufficient to show the flexibility of the *Scholar Reviews* concept. The program *STUDY* does not have to be modified in any way as long as the lesson files are formatted as *STUDY* expects them.

LESSON FILE PREPARATION

The most critical part of the successful running of the

```

16 CLS:PRINT@45,"Welcome to "R$" Scholar
's "N$" reviews!":PRINT@86,"Multiple Cho
ice or True/False"
17 C=0:PRINT@162,"Which lesson do you wi
sh to review?":PRINT@202,E$CHR$(75):;LIN
EINPUTF1$:IFF1$="ORLEN(F1$)>6ORMID$(F1$
,2,1)=":THENPRINT@247,R$ Try again, fi
lename only! "N$;GOTO17
18 PRINT@240,E$CHR$(75):;PRINT@252,"[R]a
m or [D]isk ";
19 D$=INPUT$(1):D$=CHR$(ASC(D$)AND223):I
FD$<>"D"ANDD$<>"R"THEN19
20 IFD$="D"THENF$="0":"+F1$+" DO"ELSEF$=F
1$
21 OPENF$FORINPUTAS1:INPUT#1,X$,Q,T$:IFX
$<>Z$THEN44
22 CLS:PRINT@47,"Scholar is ready to Rev
iew:":PRINT@((40-LEN(T$))/2+120,R$) "T$"
"N$":FORI=1TOQ:INPUT#1,A,X$,Q$:IFX$<>Z$AN
DX$<>T$THEN44
23 IFX$=Z$THEN32
24 FORJ=1TO4:INPUT#1,A$(J):NEXT:INPUT#1,
V,V$,PV$:GOSUB11:CLS:PRINT@41,Q$":":PRIN
T:FORK=1TO4:PRINT "RIGHT$(STR$(K),1)"
"A$(K)
25 NEXTK:PRINT@281,R$"? "N$ " ";
26 A$=INPUT$(1):AN=VAL(A$):IFASC(A$)=27T
HENCLOSE:GOTO43
27 IFAN<1ORAN>4THEN26
28 IFAN=ATHENGOSUB12:GOTO30
29 GOSUB13:GOTO26
30 IFVTHENPRINT@((40-LEN(A$(A)))/2+120,A$
(A):PRINT@((40-LEN(V$))/2+200,V$:GOTO39
31 PRINT@((40-LEN(Q$))/2+120,Q$:PRINT@((40
-LEN(A$(A)))/2+200,A$(A)".":GOTO39
32 IFA=1THENAN$="F"ELSEIFA=2THENAN$="T"E
LSE44
33 INPUT#1,V,V$,PV$:GOSUB11:CLS:PRINT@81
,Q$":":PRINT@171,"[T]rue or [F]alse?":PR
INT@219,R$"? "N$ " ";
34 A$=INPUT$(1):A$=CHR$(ASC(A$)AND223):I
FASC(A$)=27THENCLOSE:GOTO43ELSEIFA$<>"T"
ANDA$<>"F"THEN34
35 IFA$=AN$THENGOSUB12:GOTO37
36 GOSUB13:GOTO34
37 IFVTHENPRINT@((40-LEN(V$))/2+120,V$". "
:GOTO39
38 PRINT@((40-LEN(Q$))/2+120,Q$". "
39 IFI<>QTHEN41
40 FORT=1TO2000:NEXT:CLOSE:GOTO42
41 NEXT
42 CLS:PRINT@243,"Review another lesson?
":A$=INPUT$(1):IFA$="Y"ORA$="y"THENCLS
:GOTO17
43 CLS:PRINT@81,"Without use of reviews,
you were able to get "C"out of "Q"corre
ct.":FORI=1TO40:POKE64703+I,229:NEXT:FOR
I=41TO80:POKE64703+I,231:NEXT:FORI=81TO2
71:POKE64703+I,P(I):NEXT:FORI=281TO320:P
OKE64703+I,230:NEXT:CLS:CALL21795:CALL16
974:MENU
44 CLS:PRINT@41,R$ "F$ " "N$ " is not pro
perly formatted":PRINT and, very likely
, is not a "R$" Scholar "N$" file, wh
ich requires a specific format.":;PRINT:

```

Continued.

Scholar Reviews program is typing in the lesson data carefully, noting especially where you place commas and double quotation marks. Missing or misplaced characters can cause the program to terminate prematurely.

The first line of any lesson file is different from the rest of the lines in the file. It tells *Scholar Reviews* essential information about the lesson and *must* contain the following three items separated by commas:

- l - Essential *Scholar* identifier, tells the program that this is a lesson file
- n - The number of questions in the lesson
- xxx - The title of the lesson—no more than thirty-eight characters

For example, *l,75,Olympic Games* indicates that the lesson file on the Olympic games contains seventy-five questions.

After the first line, each record (i.e., each lesson question) must contain either ten or six items (depending on type of question—multiple choice or true/false) separated by commas. If an item contains textual material (a question or response, for example) and a numeral (1, 2, 3, etc.), you should enclose it in double quotation marks. Always enclose the last item, the response to an incorrect answer, in double quotation marks (").

★ Missing or misplaced characters can cause the program to terminate prematurely. ★

Here is the format for multiple choice questions:

- 1) n - The number of the answer to the question (1 to 4)
- 2) l - Essential *Scholar* identifier; indicates that the question is multiple choice
- 3) xxx - The question; no more than 38 characters
- 4) xxx - Possible answer, no more than 35 characters
- 5) - Second possible answer
- 6) - Third possible answer
- 7) - Fourth possible answer
- 8) n - Type-of-response identifier for correct answers; 0 if no change in response, 1 if the response is rephrased
- 9) xxx - Alternate response phraseology if item 8 is 1 and the answer is correct; use an arbitrary single letter if item 8 is 0
- 10) "xxx" - Text of the information to be displayed when an incorrect answer has been chosen; may be up to 160 characters including spaces; always enclose in double quotation marks

Here are some examples of multiple-choice questions as

```
PRINT" Be sure to see Scholar documentat
ion.":PRINT@280,P$;
45 A$=INKEY$:IFA$=""THEN45ELSEMENU
46 IFERR=54THEN44
47 IFERR=52ORERR=55THENCLS:PRINT@41,"Sor
ry, unable to locate "R$" "F$". "N$:PRINT
@81,"File MUST be a .DO file in Scholar'
s format. See SCHOLAR DOCUMENTATION."
:PRINT@211,"Try again? (y/n) ";:A$=INPUT
$(1):IFA$="Y"ORA$="y"THENRESUME15ELSEMEN
U
48 IFERR=18ORERR=68THENCLS:PRINT@41,"Dis
k drive not on, or DOS not loaded.":PRIN
T:PRINT" Is disk in the drive?":PRINT:PR
INT"Press F4 or type RUN <Enter> to rest
art":END
49 CLS:PRINT"Error"ERR" in line"ERL:END
```

End of listing.

typed into the lesson file:

3,l,Which listed is a continent?,United States of America,Washington D.C.,Africa,Great Britain,1,is one of the continents.," The seven continents of the world are North and South America, Asia, Africa, Europe, Australia, and Antarctica."

In the above, the 1 following *Britain* indicates that the response to a correct answer will use different phraseology from the question.

2,1,John Adams was the nation's first,Treasurer,Vice president,President,Chief justice,0,a," Later president, and father of another president, John Adams was the first vice president of the new nation."

Here the question is used again for the response phrase, so 0,a follows the word *justice*.

The following shows the format for true/false questions:

- 1) n - The number of the correct answer to the question, where 1 indicates false, 2 indicates true
- 2) ~ - Essential *Scholar* identifier indicating that this question requires a true/false answer
- 3) xxx - The question; no more than 38 characters
- 4) n - Response-type identifier for correct answer; 0 if no change in response, 1 if the response is rephrased
- 5) xxx - Alternate answer phraseology if item 4 is 1; use an arbitrary single letter if item 4 is 0
- 6) "xxx" - Text of the information to be displayed when an incorrect answer has been chosen; may be up to 160 characters including spaces; always enclose in double quotation marks

The following is an example of a true/false question:

1,~,Greenland is the smallest continent,1,Greenland is an island," The seven continents are Europe, Asia, Africa, Australia, Antarctica, North America, and South America"

Note that the tilde (~), indicating a true/false question,

is used in item 2 instead of the vertical line (|).

COMPOSING RECORDS:

Assuming that you have a ready source of questions and answers, perhaps the trickiest portion of lesson preparation is composing the last item for each question, the supplementary information. Note that the information for this item in the above examples is enclosed in double quotation marks; these quotation marks are to preserve the leading space of the item for a good format when it's printed on the computer screen. Also, the quotation marks are ESSENTIAL to the proper operation of the program for other reasons. You need the other spaces in the data to get a well-formatted display when the program is run.

```
|,16,General Knowledge
4,|,Select the European
country,Florida,Japan,Israel,Poland,1,is in Europe., "Japan
and Israel are in Asia, Poland is on the continent of
Europe, & Florida, part of the U.S., is in North America."
1,|,President of U.S. during World War I,Woodrow Wilson,Jack
Pershing,John Rockefeller,Benjamin Franklin,1,"was President
in W. W. I, 1914-1918", "Pershing, Franklin, and Rockefeller
were never President of the U.S., but Wilson served from
1913-1921."
1,~,Greenland is the smallest continent,1,Greenland is an
island," The seven continents are Europe, Asia, Africa,
Australia, Antarctica, North and South America"
4,|,Located on the U.S. south border
is,Cuba,Brazil,Canada,Mexico,1,is south of the Rio Grande.,
Canada is north of the U.S., Mexico is directly south, Cuba
is an island, and Brazil is in South America."
1,|,Which group contains only lakes,"Erie, Baikal,
Como", "Huron, Superior, Pacific", "Hudson, Huron,
Haiti", "Eire, Ontario, Atlantic",1,are all lakes., "Lake
Como is in Italy, Lake Baikal in the Soviet Union, and
Lake Erie in the United States & Canada."
3,|,Commander of the Confederate Army,Robert Louis
Stevenson,Washington Irving,Robert E. Lee,Ulysses
Grant,1,was the Confederate Commander-in-Chief," Grant
commanded the Union Army for the final phases of the Civil
War and Lee was the Southern commander. Stevenson and
Irving were authors."
2,~,Alaska is the largest state in the U.S,0,a," Texas, the
second largest state, is less than half the size of
Alaska."
4,|,The first President of the U.S. was,Abraham Lincoln,John
Quincy Adams,John F. Kennedy,George Washington,0,a," George
Washington from Mount Vernon in Virginia, 'the Father of
our country,' was also the first American President."
2,|,Which of those listed is in Asia?,Brazil,Saudi
Arabia,Algeria,New Zealand,1,is part of the Asian
continent., "Brazil is in South America, Algeria is in
Africa, Saudi Arabia in Southwest Asia, and New Zealand
is in Oceania."
3,|,The capital of Maryland
is,Baltimore,Towson,Annapolis,Anaheim,0,a," Baltimore is the
largest city in Maryland, and a major seaport, but
the capital of Maryland is Annapolis."
1,|,On which continent is Maryland?,North
America,Europe,Asia,United States,1,is Maryland's
continent., "Maryland is located in the country of the
United States, and that nation is sited on the North
American continent."
2,|,The national capital is,New York,Washington
D.C.,Chicago,Philadelphia,0,a," Though Philadelphia was an
early site for the federal government, since 1800
Washington has been the capital."
3,|,Which listed is a continent?,United States of
America,Washington D.C.,Africa,Great Britain,1,is one of the
continents., "The seven continents of the world are North
and South America, Asia, Africa, Europe, Australia, and
Antarctica."
2,|,John Adams was the nation's first,Treasurer,Vice
President,President,Chief Justice,0,a," Later President, and
father of another President, John Adams was also the
first Vice President of the new nation."
4,|,Select the capital of the
U.S.S.R.,Paris,Stalingrad,Warsaw,Moscow,1,is the Soviet
Union's capital., "Warsaw is in Poland, Paris in France,
and Stalingrad is in the U.S.S.R, but Moscow is the Soviet
Union's capital."
3,|,Franklin D. Roosevelt was President, "1812-1816", "1956-
1966", "1933-1945", "1898-1902",0,a," In office longer than
any other person Franklin Roosevelt became President on
March 4, 1933 and died April 12, 1945."
```

End of listing.

Tip: Prepare the entire record by starting in *BASIC*! Decide on the correct answer and type it, the comma, the proper identifier (1 or ~), the comma, and the question. Then use the *EDIT* command to compose the remainder of questions, answers, and review information. This procedure enables typing of precisely the format desired with no concern about spacing as can happen in *TEXT* mode. Finally, *CUT* or *COPY* the record into the paste buffer and exit the *EDIT* mode; proceed to the *TEXT* work file for the lesson and *PASTE* the record in place.

THE PROGRAM LISTING

This program depends heavily on use of the *Telcom* previous-screen buffer (also know as the alternate LCD buffer) and the fine work done by David Sumner in his article, "Overcoming the Eight-Line Barrier," in the first issue of the lamented *Laptop User* magazine (Mar/Apr, '87). The primary purpose of developing this program was to explore the practical applications of Sumner's thesis. Perhaps, with this model, others can go on to further application development.

- * Line 7 jumps to line 14 for initialization of the program.
- * Lines 8-10 contain the data for the screen display on normal exit.
- * Line 11 is the critically important subroutine that *POKE*'s the question and review information into the *Telcom* previous screen buffer.
- * Line 12 is the subroutine called by line 28 or 35 when the student selects the correct answer.
- * Line 13 subroutine calls the previous screen buffer when called by line 29 or 36 because of a student error.
- * Lines 14-20 are fairly standard initialization and welcome processes.
- * Line 21 opens the requested lesson, checks for identification, then picks up the number of questions and the lesson's title. An error trap jumps to line 44 and terminates the program if identification fails.
- * Line 22 acknowledges the program's readiness to proceed and commences the main loop by reading the correct answer, identifier, and question from the lesson. Error trapping looks for the identifier.
- * Line 23 jumps execution to line 32 if the question is true/false.
- * Lines 24-31 (for multiple choice) and 32-39 (for true/false) read the answers and tutorial material from the lesson and display them. The *GOSUB 11* places the tutorial in the previous screen buffer ready to be *CALL*ed if the student errs.
- * Lines 40-43 complete the loop, provide for further reviews, and exit the program after preparation and *POKE*ing of the exit display.
- * Lines 44-49 provide the necessary error trapping assistance.

Comments and questions may be directed to the author:
 Dave Thomas, 2308 Chetwood Circle, #103, Timonium, MD 21093, Phone: (301)252-5379. Electronic mail: GENIE, MORTIMER; Delphi, MORTIMER; CompuServe, 70526,1117

COMPATIBILITY: Tandy 100/102; Tandy 200; Kyotronic 85; Olivetti M10—untested; NEC 8201 (with changes)

A Notebook-Computer Teacher

LEARN: Tutoring and testing with a notebook computer.

By E.A. Schwartz

In a few months I have to take a test to prove I can cope in Spanish. It's either that or give up the idea of finishing graduate school. Being naturally contrary, I've decided the way to bring my Spanish up to speed is not to *take* lessons but to *write* lessons—a set of computerized lessons based on a textbook from a language school where I studied in Mexico. *LEARN* is the program I wrote to run those lessons.

Basically, *LEARN* takes prompts and questions from a .DO file, presents them to a user, and responds to the user's answers. When the answer given is incorrect, its prompt, question, and correct answer are stored in another file. When the original lesson file is exhausted, the "incorrect answer" file is opened and the lesson continues without a break till the user has answered each question

The program is set up so you can also use this test mode to make electronic forms.

correctly. The user can exit from the program by answering *end* at any question.

LEARN also has a test mode, which asks each question only once, records the answers, and calculates a score at the end. The program is set up so that you can also use this test mode to make electronic forms. For example, you could write employment application questions in a .DO file so that applicants could answer those questions at the computer. Such a form could be infinitely variable, and you wouldn't have to use paper.

WRITING LESSONS

Putting together a .DO lesson file that works with



Nupa Two Shoes takes the sample U.S. History test, coached by the author (he scored 80%).
Photo by Minnie Two Shoes

LEARN is not difficult—you can easily write your own lessons and tests. But you have some limitations. Here are the rules.

Think of a lesson as at least one group of prompts and their corresponding answers. Each prompt consists of a *string*, a sentence or group of words ending with an *ENTER*. The *ENTER* tells the program that it has reached the end of each string (so when you're creating a lesson file, be careful not to press *ENTER* in the middle of a string). See Figure 1 for an example lesson file. Each group is organized with the following parts:

1. A period signifying the beginning of a lesson group;
2. A statement (explanation for the questions) giving background information to be used in the questions;
3. A statement that tells how to answer the questions (for example, to type the answer in lower case, to type *T* or *F* for true/false, etc.);
4. At least one pair of strings—a question and its corresponding correct answer.

Each lesson group starts with a period (.) followed by an *ENTER*. This period is the first string of the group and tells the program that a new lesson group is beginning.

The next string in the group is the question explanation and is the first string of the group displayed. This string generally supplies the information needed to answer the questions that follow. In Figure 1, the statement, *The Spanish sentence "Donde esta el telefono, por favor?" means, in English, "Where is the telephone, please?"* is the question explanation string and is followed by an *ENTER*. As the program runs, the explanation is put on the screen followed by the *press enter to continue* prompt, which allows the user to study the explanation as long as needed. When the user answers a question incorrectly, the explanation flashes on the screen again before the next question.

After the explanation come the instructions. These tell the user how to answer questions—by typing a letter or number for multiple choice questions, or by typing a word to fill a blank in the question, for example.

Finally comes the set of alternating questions and their answers. In the example lesson (Figure 1), the questions ask for the meanings of each word in the sentence given in the explanation. Each element of this set of questions consists of one string (the question), with one *ENTER* (signifying the end of the string). No element may be more than 255 characters long. The sample *test* (see Figure 2) shows how to use the first element for explanations.

Although each question must be one string only (it contains no *ENTER*s in the lesson file except at the end), as an added feature the program allows you to start new lines on the computer screen in the question by putting a square bracket (I) where the new line is to start. This feature allows you to set up multiple-choice questions so that each possible answer is on its own line. Further, you can provide two possible correct answers by putting a slash (/) between them in the answer string in a learn file. Note that the program is case sensitive, so you should inform students that they should use either all upper case or all lower case.

```
.
The Spanish sentence "Dónde está el teléfono, por favor?"
means, in English, "Where is the telephone, please?"
Type in the meanings in English in lower case letters.
What does "dónde" mean?
where
What does "está" mean?
is
What does "por favor" mean?
please
What does "el" mean?
the
.
Here are some words you might use with "dónde está": "banco"
means bank, "mercado" means market, "farmacia" means
pharmacy, and "iglesia" means church.
Enter the letter that identifies the correct answer
You ask, "Dónde está el mercado?" Are you asking for (a) the
location of a church, or (b) the location of the market?
b
You ask, "Dónde está la Iglesia de San Pedro?" Are you
looking for (a) a church, or (b) a bank?
a
You ask, "Dónde está un Banco Nacional?" Do you want to find
(a) a beach, or (b) a bank?
b
You ask, "Dónde está una farmacia, por favor?" Are you
looking for (a) a sandwich, or (b) a bottle of aspirin?
b
.
"Quiero un cuarto para esta noche" means "I want a room for
tonight." (Note that "esta" is not "está.")
Enter the letter indicating the correct answer.
Does "esta noche" mean (a) "a room," or (b) "tonight"?
b
Does "quiero" mean (a) "a room" or (b) "I want"?
b
Does "para" mean (a) "a room" or (b) "for"?
b
Does "un cuarto" mean (a) "a room" or (b) "I want"?
a
```

Figure 1. A sample lesson with questions and answers. The program handles multiple choice, fill-in-the-blank, or standard question-and-answer formats.

The program knows it has reached the end of a lesson group when it comes either to another period—to signify the beginning of the next group—or to the end of the file.

THE STUDENT'S PERSPECTIVE

For the student running the program, this is what shows on the screen. For any lesson group, the first string, the explanation, comes on the screen and stays until the student presses any key. Once a key is pressed, the second string, the instructions, and third string, the first question, come on the screen together and remain until the student types a response (normally an answer to the question) and presses *ENTER*. If the student's response doesn't match the correct answer, the string following the question in the lesson file, the explanation string, is flashed on the screen again. Either way, the second string (the instruction string) does not come on again until either a new lesson group begins (the program comes to a period by itself on a line) or the program reaches the end of the lesson file and begins drawing on the "incorrect answer" file.

This format obviously does not lend itself to essay questions, but it works well with multiple-choice, true/false, and fill-in-the-blank questions.

TEST MODE

The test mode has some primitive but, I hope, effective security gadgets attached. The person giving the test should set it up by typing a lower-case *t* (case is important; uppercase won't register) in response to the first display of the program, the first question explanation. Then the tester types in the test file name. Suppose, then, that the

testee is somewhat computer literate and decides the answers to the questions must be somewhere in the computer. He or she may break from the program, discover the answer file (the test file), and try to get back into *LEARN*. But nothing in the first display suggests how to get into the test mode. If the person taking the test presses only *ENTER* and goes through the test sequence, no new answer file (named *RSPONS.DO*) will be created. If the testee discovers what a *RSPONS* file looks like and tries to construct one outside the program, he or she probably won't know to put the score and the time at the bottom of the file where the program would put them. The time is a tip-off if the testee dilly-dallied after completing the file. (Before doing any dilly-dallying, s/he would have to know that after *Test Completed* flashes on the screen, you must press lower-case *e* to get back to the menu.)

In addition to the score, the program prints all questions and answers in *RSPONS.DO*. Incorrect answers are marked by a black box (*CHR\$(239)*).

The test mode also allows you to ask questions that don't have correct answers—the name of the person being tested, for example. To ask questions that won't be counted in the score, put *ok* where the correct answer would otherwise be located in the test file.

TIMING

The program lets you adjust how long the explanation remains on the screen after an incorrect answer. The initial display tells the you to press *a* to adjust timing. That routes the program to the adjustment sequence. (The time span secondary explanations stay displayed depends on the length of those explanations.) Since this prompt is at the beginning of the program, and the instructor must enter *t* to start the test mode legally, there's no way for the student to set the speed. It must be done by the instructor.

VARIABLES

- A Timing adjustment factor
- A\$ Answer
- B Indicates the current stage of the prompt-loading.
- C\$ Dummy for *INKEY\$* sequence
- D If *C* equals 1, the file *K.DO* has been opened for output
- E If *D* equals 1, the period, explanation, and instructions have been printed in the "incorrect answer" file
- E *E* equals 1 indicates that the program has finished the first "incorrect answer" file
- E\$ Explanation
- F Print formatting sequence search starting position
- F\$ File name
- I\$ Instructions
- K\$ Replacement "incorrect answer" filename in lesson mode
- L Position of space in the print formatting sequence
- L\$ Name of "incorrect answer" file in lesson mode
- N Position of the square bracket (*I*) used to start a new line in the question display
- Q\$ Question
- R\$ Second of two possible answers
- R Number of right answers in test mode
- S\$ Basic input string, or first of two possible answers

```

.<
This is a 10-question multiple-choice quiz on basic U.S.
history.<
Press enter to continue.<
ok<
.<
Your name?<
ok<
.<
Enter the number of the correct answer.<
The Pilgrims landed in Massachusetts in 1620 to found which
colony, (1) Boston, (2) Jamestown, or (3) Plymouth?<
3<
In April _____, the American Revolution began at Lexington.
In the following year, the Declaration of Independence was
signed. Identify the year that belongs in the blank space:
(1) 1763, (2) 1775, or (3) 1781.<
2<
In 1803, the U.S. bought Louisiana from France. Which
French leader sold Louisiana, (1) Louis XIV, (2) Napoleon,
or (3) Napoleon III?<
2<
Which U.S. president, in 1835, forced most of the Cherokee
Indians to give up their homes in the southeastern U.S. and
march the Trail of Tears to Oklahoma, (1) James Monroe, (2)
Martin Van Buren, or (3) Andrew Jackson?<
3<
In May _____, the U.S. declared war on Mexico. That war
ended two years later, in the year when gold was discovered
in California. What year belongs in the blank, (1) 1861,
(2) 1847, or (3) 1846?<
3<
Fighting in the Civil War began in 1861 with the bombardment
of a fort in the harbor of what city, (1) Charleston, (2)
Hampton Roads, or (3) Norfolk?<
1<
The U.S. bought Alaska in 1867 for $7.2 million from what
country, (1) Spain, (2) Great Britain, or (3) Russia?<
3<
In 1890, some 200 Indian men, women, and children were
killed in the "battle" of Wounded Knee. What tribe was
involved, (1) the Kiowa, (2) the Lakota Sioux, or (3) the
Seneca?<
2<
Two U.S. presidents are known to have been assassinated
during the 20th century. They are (1) Kennedy and McKinley,
(2) Kennedy and Garfield, or (3) Harding and Taft?<
1<
The U.S. entered World War I and passed the prohibition
amendment in which year, (1) 1914, (2) 1917, or (3) 1920?<
2<

```

Figure 2. A simple multiple-choice quiz on U.S. History (The same one taken by Nupa Two Shoes).

- T *T* equals 1 indicates test mode
- T\$ Dummy for *INKEY\$* sequence
- U Length of *U\$*, timing adjustment
- U\$ String to be processed through the print-formatting and timing.
- V Timing sequence dummy
- V\$ Portion of *U\$* extracted during the print formatting.
- W Number wrong in test mode
- X *X* equals 1 indicates that the last sub-string of *U\$* has been processed in the print-formatting sequence
- X\$ Period indicating a new question group is beginning
- Y Position of slash (/) in the answer string to allow for two correct answers
- Z *Z* equals 1 indicates that the program is running the "incorrect answer" file
- Z\$ Answer given by user

FUNCTION LOCATIONS

- 100-110 Prints the initial display, opens the initial file(s).
- 115-150 Inputs file data, assigns it, and routes display data to word-wrap and timing subroutines.
- 155-200 Compares the user's answer with the correct answer from the lesson file; responds to an in-


```

100 CLEAR1200:MAXFILES=2:A=7:K$="k.do":L
$="1.do"
102 CLS:PRINT"LEARN":PRINT"Press enter t
o continue":PRINT"s to stop or":PRINT"a
to adjust timing":GOSUB290:IFT$<>CHR$(1
3)THEN102
105 ONERRORGOTO315
110 CLS:INPUT"File";F$:F$=F$+".do":OPENF
$FORINPUTAS1:IFT=1THENOPEN"rspons.do"FOR
OUTPUTAS2
115 LINEINPUT#1,S$:IFS$="."THENX$=S$:B=1
:D=0:GOTO115
120 IFB=1THENE$=S$:B=2ELSE140
125 IFT=1ORE$=" "THEN115
130 CLS:U$=E$:U=LEN(U$):IFZ=1THENGOSUB27
0:GOSUB250ELSEGOSUB270:PRINT@240,"(press
enter to continue)":GOSUB230
135 GOTO115
140 IFB=2THENI$=S$:U$=I$:B=3:CLS:IFU$=" "
THEN115ELSEU=LEN(I$):GOSUB270:GOTO115
145 IFB=3THENQ$=S$:U$=Q$:B=4:GOSUB270:GO
TO115
150 IFB=4THENA$=S$:B=3
155 LINEINPUTZ$:Y=INSTR(1,A$,"/"):IFY>0T
HENS$=LEFT$(A$,Y-1):R$=RIGHT$(A$,LEN(A$)
-Y)
160 IFT=1THEN255
165 IFZ$="end"THEN320
170 IFZ$=A$ORZ$=R$ORZ$=S$THEN195ELSEU$="
Incorrect":GOSUB245:CLS:IFEOF(1)THEN180
175 U$=E$:U=LEN(U$):GOSUB270:GOSUB250:CL
S
180 IFC=0THENOPENK$FOROUTPUTAS2:C=1
185 IFD=0THEND=1:PRINT#2,X$:PRINT#2,E$:P
RINT#2,I$
190 PRINT#2,Q$:PRINT#2,A$:GOTO200
195 U$="Correct":GOSUB245:CLS
200 IFEOF(1)THEN205ELSE115

```

Continued.

```

205 IFT=1THENPRINT#2,"Score:"INT(((R/(W+
R))+.005)*100)"Pct "TIME$:PRINT"Test Com
pleted":GOTO235
210 IFE=1THENCLOSE1:KILL$
215 IFC=1THENC=0:Z=1:CLOSEELSE225
220 NAMEK$ASL$:E=1:OPENL$FORINPUTAS1:GOT
O115
225 CLS:U$="Lesson completed":GOSUB245:M
ENU
230 C$=INKEY$:IFC$=" "THEN230ELSERETURN
235 T$=INKEY$:IFT$=" "THEN235
240 IFT$="e"THENMENUELSE235
245 PRINTU$
250 FORV=1TOU*A:NEXTV:RETURN
255 PRINT#2,Q$:IFA$="ok"THEN265
260 IFZ$=A$THENR=R+1ELSEW=W+1:Z$=CHR$(23
9)+Z$
265 PRINT#2,Z$:CLS:GOTO200
270 N=INSTR(1,U$,"["):IFN=0ORN>39THEN275
ELSEV$=LEFT$(U$,N-1):U$=RIGHT$(U$,LEN(U$)
-N):GOTO285
275 F=1:IFLEN(U$)<40THENV$=U$:X=1:GOTO28
5
280 L=INSTR(F,U$," "):IFL>39ORL=0THENV$=
LEFT$(U$,F-1):U$=RIGHT$(U$,LEN(U$)-(F-1)
)ELSEF=L+1:GOTO280
285 PRINTV$:IFX=1THENX=0:RETURNELSE270
290 T$=INKEY$:IFT$=" "THEN290
295 IFT$="s"THENMENU
300 IFT$="t"THENT=1
305 IFT$="a"THENCLS:GOTO310ELSERETURN
310 INPUT"higher #s = slower speed (defa
ult 7)":A:RETURN
315 IFERR=15THENZ$="":CLS:PRINTQ$:RESUME
155ELSEPRINTERR"in"ERL:STOP
320 IFC=1THENKILLK$
325 IFE=1THENKILL$
330 MENU

```

End of listing.

LEARN.BA, a powerful tool for memorizing materials and testing students.

- correct answer by storing data in the replacement file and flashing an explanation.
- 205-225 End-of-file sequence, declares the replacement "incorrect answer" file as the main file.
- 230-240 INKEY\$ sequences.
- 245-250 Timing subroutine.
- 255-265 Test mode answer response sequence.
- 270-285 Word-wrap and printing subroutine.
- 290-310 Subroutine for response to the initial display.
- 315 Error response.
- 320-330 Response to the "end" answer.

Editor's: Because the test mode is programmed a little differently from the lesson mode, it asks unexpected questions in the beginning sequence, and (for example) decides that the ENTER

key is "Correct," while your name is "Incorrect." The program is case sensitive we suggest that you carefully explain to the student users how you want them to type upper- and lower-case letters.

NEC Changes:

Add to end of line 100—:SCREEN0,0
 Line 102—change "enter" to "return"
 Line 130—change PRINT@240,"(press enter to continue)" to LOCATE0,6:PRINT"press return to continue"
 Either define any necessary foreign language graphics characters (see your BASIC manual), or don't use any and instead, change CHR\$(239) in line 260 to CHR\$(130).

A Dandy Teacher's Aid



Grading can be made easier with
Steffens' Grade Management System



by David Klein

No teacher alive likes to figure end-of-term grading. Not only are you tired from weeks of teaching (and you're looking forward to that long-deserved vacation), you are staring face-to-face with piles of last-minute papers or finals or projects (or all the above), and you certainly have no time or inclination for dragging out your calculator to wrestle with the complex grading formula you devised at the beginning of the term when your mind was rested. If you're like me, you've been looking for a program that can help you keep track of grades *during* the term and will take care of calculations at the end of the term. The Grade Management System (GMS) by Software by Steffens', a program made for Model 100's with disk drive, may get you close.

Every teacher I know has a different way of keeping grades—different from anyone else who ever breathed. So a software grading package, especially for notebook computers, faces an enormous task when it tries to appeal to a large contingent of teachers. Dauntlessly attempting that, in the introduction to the software, the authors state these two assumptions: 1) teachers want to keep their handwritten grade books because they fear loss of valuable data, but 2) teachers nevertheless want to use computers to calculate grades quickly and easily, as long as the

computers are as easy to use as a grade book.

I absolutely agree with the second assumption. But as a person who has spent, unsuccessfully, upwards of sixty to eighty hours trying to get a Lotus clone to do my grading, I gener-

**It tracks
students' daily
performance.**

ally disagree with the first. Ideally, for me, the optimum grading program allows me to type in grades as each assignment comes due, keeps track of each assignment through the term, permits easy changes in grade weight, and allows frequent changes in grades already entered. Rather than worry about losing my valuable data from some computer accident, I want to print an updated version of my grading sheet every time I type in grades for an assignment, ensuring plenty of copies, instead of a tenuous, marked-up, smudged, dog-chewed (yes, it happens to teachers, too) paper copy. Too often have I experienced the cold sweat of searching for an irreplaceable piece of paper, on which is recorded part of the lives, the several weeks of hard work, of

my students.

The GMS doesn't go that far. But it comes close. The system keeps track of the accumulation of grades for a term—not each assignment—as well as a miscellaneous recording of daily notes, such as absences, tardiness, late or forgotten assignments—remembered assignments—and the like, whatever may determine the final grade for each student. The program is flexible enough that it can be used by teachers from elementary to college level. It allows teachers to separate their subjects or classes and the students' performance in them. It tracks students' daily performance. It keeps a file of delinquent assignments. And it provides several convenient ways of compiling and printing this data.

But most important, the GMS permits you to keep this data on 3.5-inch disk in Model 100 format. If you want, you never have to bring your data files into RAM with an explicit command. The programs take care of much of the significant data handling on the disk, which allows you to keep a data base of grades for up to fifty students in as many as nine subjects or classes. Not bad for a Model 100. Further, the programs are easy to learn, asking enough questions and giving enough warning so that a neophyte user can manage without having to refer constantly to the instructions. Although, at times I felt that some processes could require

fewer precautions, say, with a default and *ENTER* included as a response instead of a particular key such as *Y* or *N*, which you have to hunt for.

Speaking of instructions, here's the bad news. Especially for a manual designed to be read by teachers, the instruction booklet that comes with the program is an indication of why some very good programs don't make it on the market. It has typos, spelling and grammar errors that might slow up those who are English teachers (such as I). But I must mention two major problems. For me, the writing was good enough to get me started, but often the writing in the instructions became somewhat opaque, frustrating me until I finally gave up on them and turned to the program itself to teach me how to use it. Consider these: *The program menu appears to follow numerically.* (to follow what?) *User inputs (for example assign. notion) "P.384" press ENTER.* If section "2" was made screen. Yes, these are punctuated as complete sentences. Being a teacher himself, Gary Steffens has a limited budget and time, and I'm sure the deficiencies in the instructions are a result of this.

The second problem with the documentation is the presentation of the material. The documentation does not focus on the sample files included in the disk, which explain reams' worth of instructions. But more confusing, the ordering of the tutorial is not optimal. I started the tutorial on a Model 102 with 24K of RAM, but soon discovered that, for lack of space, I couldn't run many of GMS's programs with the data files in RAM. But the instructions explicitly warned against using the disk drive. I thought I was stuck. However, I moved to a Model 100 with 32K of RAM and became familiar enough with the program to discover that using a disk was remarkably easy. I could have started using the disk from the beginning if the trick of naming files had been explained earlier (the program permits you to append numerals—a 1 for the first term, 2 for the second, etc.—to otherwise identical file names to permit continuity among school terms and to allow you to use the same class lists from term to term). If you try the

program, I advise you to create some dummy files to play with, and use the disk drive at the outset.

For a program designed for the Model 100 where size is a consideration, this program is surprisingly versatile. Of course, you may find yourself somewhat constrained by its grading assumptions. For example, the program gives two kinds of graded assignments only—classwork and tests—for which it assigns a final, cumulative grade based on a fifty-fifty split (fifty percent counted for classwork and fifty percent for tests). But if you're like me, unless you write your own, no program is going to satisfy all your requirements. On the other hand, the program is written in *BASIC*, so many parts (such as the fifty-fifty formula or the printouts) may be relatively easy to alter, although the instructions recommend that you do not alter the program (a good idea if you want manufacturer's support). Or if



you are loathe to do your own programming, the people at Software by Steffens' will alter the program for you, ten dollars for easy changes, and twenty an hour for more extensive ones. Decent prices.

Further, the program allows you to choose your own grading scale, such as 90 to 100 as a range for an A and so on (although if you use the 4.0 scale, I advise you to multiply your range by 10, since the program doesn't handle fractions well). It gives you flexibility with names and number of data files you need. It gives you exceptional ability for tracking individual students—probably the second best feature of the program, after the use of the disk. It pinpoints students who haven't completed assignments or who have missed tests (here's a way to help prevent those students from always falling through the cracks). And most

important, it permits you to carry your grades in a Model 100. Suddenly you have access to a data base that makes your grading sheet three dimensional, allowing you to view your grading records from a variety of perspectives, class averages, class totals, numbers of A's, B's, etc., and as end-of-term reports. You don't have to leave your computerized grade book at home strapped to your desk, and you don't have to recalculate your grades at term's end. Press a few buttons and that's it.

Should you try the program? If you are looking for a way to ease the end-of-term crunch by having your grades already *number-crunched* with a computer, this program could well be worth the investment at \$49. You can use it immediately to enhance your grading sheet and to incorporate a little sanity in your grading procedure. Perhaps Gary and Lee Steffens will produce an enhanced version that lets you put your entire grading sheet on-line (perhaps providing an easier way to change data, after it's been entered, when you discover you are not perfect at data entry). Coupled with the program as it stands, such an enhancement would make GMS a teacher's dream. For me, a computer-oriented person who has *never* used a computer for grading because I have never found one versatile enough and haven't had time to write one, GMS does the job. The payoff would be worth it.

David Klein has been teaching Freshman English at the University of New Mexico.

Manufacturer's Specifications

Grade Management System
\$49.00

Software by Steffens'
1731 William Ct.
Little Chute, WI 54140

Requirements:
Model 100/102/200
3.5-inch disk drive
Enough knowledge to *LOAD* programs from disk and to initiate a program in RAM.

Is LapDOS II Good to Be True?

A Review of Traveling Software's LapDOS II for the Tandy 1400LT

by Mike Nugent II

If you transfer files between your PC and your notebook computer, you're going to like this. You can forget about TELCOM programs, transfer protocols, binary-to-ASCII conversions, and all that other stuff. Life just got simpler.

Mine did. My work is a maze of file transfers. We get manuscripts on a variety of media and in all sorts of word processor and disk formats. Yet they're all destined for the same few places and forms: to *Microsoft WORD* for editing; Tandy PDD-1 (portable disk drive, 100K format) disks for sale; to all our different laptops for testing, modification and debugging (and occasionally for swearing at); and to the PICO BBS for downloading. Any way we get them, it's a lot of work.

LapDOS II makes it easy by tying it all together in one package. It talks to everything—Tandy and NEC notebook computers, Tandy PDD-1, PDD-2, Brother FB100, and Purple Computing portable disk drives—with no hassle. Just connect an IBM PC-compatible computer (or just PC) to a portable disk drive or notebook computer, and start swapping files. Menu-driven and environment-sensing, *LapDOS II* handles the details automatically.

WHAT IS *LapDOS II*?

This review describes Version 1.42 of Traveling Software's *LapDOS II* operating on a Tandy 1400LT providing-you-have-a-big-lap laptop computer. As the manual states, *LapDOS*

II consists of two components: 1) *LapDOS*, which transfers files between a PC and a PDD, and 2) a new program which transfers files between a PC and a notebook (i.e., a notebook computer).

The package comes with a null modem cable, a cable converter, a user manual, and a 5.25-inch diskette with the *LapDOS II* software and a *README* file of recent updates to the manual. For 3.5-inch diskettes, you can copy the files to your disk. The doc's tells you how to copy *LapDOS II* to both hard disk and floppy disk.

PDD FILE TRANSFERS

Using *LapDOS II* with a PDD is identical in most respects to using it with a notebook. *LapDOS II* is easy. Connect the PDD's cable to the PC's RS-232C cable, using *LapDOS II*'s cable converter. Turn the PDD on and insert a disk. At the DOS prompt, type *LAP* and press *ENTER*. The program loads and the screen displays a menu. The top line displays the program name, version number, and copyright notice. Below that are the two *LapDOS* windows, the MS-DOS window on the left and the PDD window on the right. The top line of each window identifies the window, displays the currently logged drive (e.g., A: for MS-DOS and always P: for the PDD), and the number of free bytes on each drive. Below that, each window lists the file names (up to forty files in two columns) on each drive and their sizes. The bottom of the MS-DOS window shows the current di-

rectory path, while the bottom of the PDD window shows the COM port in use and the baud rate selected. When referring to a Tandy PDD-2, it also displays the current bank as Bank 0 or Bank 1. Below the menu is the message line, which displays prompts, warnings and error messages when necessary. And below that, on the bottom line of the screen, is the command line listing all the available commands.

A bar cursor highlights the first file in the MS-DOS window. The arrow keys move the cursor from one file name to another and from window to window. The *HOME* and *END* keys move the cursor to the first and last file names, respectively, within a window. *PGUP* and *PGDN* move between pages of files in the MS-DOS window (when they don't all fit on one page) and between PDD-2 banks in the PDD window, provided a PDD-2 is connected. *CTRL-HOME* always returns the cursor to the first file of the first page.

A big feature of *LapDOS II* is its ability to determine automatically which type of drive you're using and to set the baud rate accordingly. Some PDD's, like the Tandy PDD-1 and the Purple Computing Disk Drive, have switches you can set to change the baud rate at which they operate. Others do not. Tandy's PDD-2, for example, has no switches and is fixed at 19,200 baud. The Brother FB100 is fixed at 9,600 baud. When *LapDOS* first tries to log the P: drive, it begins at 19,200 baud. If it

can't establish communication, it changes to 9,600 and tries again, all without you pressing a key. If you change to a different PDD, no problem; *LapDOS II* takes it in stride.

COMMANDS

There are two ways to execute any of the twelve commands on the command line. One way is to press the first letter of the command name. Alternatively, you can press *ESC*, which highlights the first command with a bar cursor, then move the cursor to the desired command, using the arrow keys or space bar, and press *ENTER*. Some commands operate on the file currently selected in one of the windows, while others require only that the cursor be anywhere within the appropriate window, and still others are not at all dependent on the cursor position. Here's a summary of *LapDOS II* commands:

Help—Displays a help menu.

Quit—Exits *LapDOS*.

Log—Changes disks.

View—Only printable ASCII characters are displayed. Fine for text files. The .BA and .CO files look rather weird.

Copy—Files are copied exactly as they are stored; that is, .BA and .CO files are transferred without being changed to ASCII. Thus a .CO file copied from MS-DOS to PDD will load and run in your notebook as a .CO file with no massaging. *Copy* lets you change the destination file name. When copying to the PDD, *LapDOS* automatically shortens file names longer than six characters and substitutes a .DO extension for any source file with an extension other than .BA or .CO. When copying from PDD to MS-DOS, it substitutes a hyphen for any illegal (for MS-DOS) characters. *Copy* prompts you if the destination already contains a file of the same name and lets you overwrite it or abort the copy. On the MS-DOS side, it can copy one file to the same drive (with a different file name, of course) or to another MS-DOS drive. It won't do it on the PDD side, however.

Wildcopy—A real time saver! I copied two complete PDD-2 disks to one MS-DOS disk in no time at all just by designating *.* as the source,

pressing *ENTER*, then twiddling my thumbs while it copied all the files quickly and smoothly. It copies one bank at a time, so I had to do it four whole times! (Tough job, eh?) One 720K, MS-DOS disk holds data from three PDD-2

disks, and then some. A fine way to archive your notebook files. By the way, this program also prompts you if the destination disk already has a file of the same name.

Format—Takes about 1 minute and 45 seconds, automatically determining the proper format according to the type of PDD it sensed. It doesn't warn of data on the disk, but it does wait for you to press *ENTER*. As a precaution, you can use the *Log* command to check the files on the disk before formatting. This command does not format MS-DOS disks.

Erase—Always prompts you before erasing, giving you a chance to change your mind.

Rename—Be careful here. If assigning a new name to a PDD file, you must include the .BA or .CO extension. Otherwise, the program assumes a .DO extension.

Goto—This searches forward from the cursor for any file name you specify, or for a group of file names if you use wildcards. It remembers the last search string, so you can search for the next occurrence just by pressing *G*, then *ENTER*. Or you can edit the search string.

Xchange—Lots going on here. This option lets you copy files from *ThinkTank* to *IDEA!* format and vice versa, or from either of those formats to ASCII. And you can copy some word processor formats to ASCII. Finally, you can copy *Sidekick Appointment* files to *IDEA!* format, but not the reverse. I found that the *Sidekick-to-IDEA!* conversion worked well, but not so with *ThinkTank-to-IDEA!* conversion. Not having *ThinkTank*, I first converted an *IDEA!* file to *ThinkTank* format, then tried to

LAPDOS (1.42), Copyright 1987, Traveling Software Inc.											
MS-DOS Drive (A:)				256000 Free				Portable Drive (P:) 12800 Free			
COMMAND	.COM	23612	ANSI	.SYS	4963	ADRS	_DD	454	ART/06.DD	10108	
APPEND	.COM	1725	ASSIGN	.COM	1523	BAUDER	.BA	56	BLKJAK.BA	8686	
ATTRIB	.EXE	8234	BASIC	.EXE	7312	CAL	_BA	23	CAL	.CO	3389
BASICA	.COM	969	CACHE	.COM	7871	CHANGE	.BA	4058	DASM	.BA	3858
CHKDSK	.COM	9819	CONFIG	.SYS	78	DVRAK	.BA	3716	FIG 1	.DD	358
CPANEL	.COM	8737	CP	.COM	2608	FIG 2	_DD	643	FIG 3	.DD	715
DEBUE	.COM	15786	DISKCOMP	.COM	3947	FIG 4	_DD	506	FMCPRT	.BA	1855
DISKCOPY	.COM	4139	DISKOPT	.COM	4611	FRAZY	.BA	3480	FRAZY	_DD	4322
DRIVER	.SYS	1102	EDLIN	.COM	7495	FSMDD	.BA	349	GLDCRD	.DD	12322
EXE2BIN	.EXE	3050	FC	.EXE	14446	HOTBOX	_DD	492	HOTBX	.CO	52
FIND	.EXE	6403	FORMAT	.COM	1188	HOTBX	_DD	129	HOTKEY	.BA	4049
GRAFTABL	.COM	1201	GRAPHICS	.COM	5546	INVISIO	.BA	802	JSDSON	.BA	355
JOIN	.EXE	8942	KEYB	.COM	1340	LESSON	.BA	75	MEMO	.BA	1432
KEYBGR	.COM	1340	KEYBIT	.COM	1340	NOTE	_DD	692	PDIPOS	.CO	5176
KEYBSP	.COM	1340	KEYBUC	.COM	1340	PDZDOS	.CO	5510	PEEKIN	.DD	7271
LABEL	.COM	2889	LF	.COM	293	POPEYE	.BA	145	RACER	.BA	1599
LIB	.EXE	24138	LINK	.EXE	39172	RACER	_DD	7627	SGDG61	_DD	14248
LPDRVR	.SYS	3452	LPSETUP	.COM	6116	STOMDD	.BA	3274	STOMDD	_DD	4000
MODE	.COM	7298	MSA	.COM	282	STU	.BA	574	STU	.BA	4000
MOUSE	.COM	11846	MOUSE	.SYS	11458	XBASIC	_DD	6053	ASM	_DD	95
= A:\				Com1:19200 Bank 0 =							
Help Log View Copy Wildcopy Format Erase Rename Goto Xchange Setup Quit											

The LapDos II MS-DOS operating screen and menu.

convert that back to *IDEA!*. The program just locked up, necessitating a warm restart. Twice. Traveling Software said they hadn't heard of the problem before and is looking into it.

Setup—This has several subcommands: "Colors"—Lets you set screen colors quickly and simply, using the arrow keys to view directly the results of changes; "Baud"—Lets you override the default baud rate setting; "Port"—Lets you specify the serial port you choose for *LapDOS* (Com1 or Com2); "Snowplow"—Removes the "snow" that sometimes appears on color monitors. It tends to slow the 1400LT's screen response significantly; "Originals"—Sets all these parameters back to their original factory defaults; "Record"—Records these settings right now, rather than at exit time. Originally this was used with the now-discontinued background memory resident option. It's rather redundant now, since any *Setup* changes are automatically recorded upon exiting *LapDOS* anyway; "Quit"—Quits *Setup*.

Some setup commands can be included in the command line that invokes *LapDOS II*, by adding them as switches (e.g. *LAP/B* defaults to 9,600 baud operation on entry). Most commands abort with *CTRL-Q*, but *ESC* seems to work as well.

NOTEBOOK FILE TRANSFERS

LapDOS II works with a notebook by treating it just like a PDD. For this to work, the notebook must ACT like a PDD. That's where *LOADER.COM* (on the *LapDOS II* program disk) comes in. First connect the PC to the notebook, using the null modem cable. At the DOS prompt, type *LOADER* and press *ENTER*. On the

notebook, in *BASIC*, type *RUN"COM88NIENN"* (for Tandy notebooks, different for NEC, but *LOADER.COM* shows on the screen what to type) and press *ENTER*. *LOADER.COM* then shoves a program called *LAPDOS.BA* into the notebook and runs it. *LAPDOS.BA* makes the notebook look just like a PDD to *LapDOS II*. Voila! Thereafter, you won't need *LOADER.COM* anymore, as long as you keep *LAPDOS.BA* in the notebook.

From then on, to swap files with the notebook, just connect the computers together, run *LAPDOS.BA* on the notebook, then run *LAP.EXE*, as before, on the PC. The same menu appears as before, but now the notebook is acting as drive P:. The files in the PDD window are the files in the notebook, and the free memory displayed is the free memory in the notebook's RAM. Commands operate the same as if a PDD were attached. Almost.

RAM files offer potential pitfalls not possible with disk files. For one, RAM files can be invisible. (Try THAT on a disk!) To *LapDOS'* credit, it handles invisible files just fine; they show on the menu and can be copied, renamed, etc., no problem.

Where you can get into trouble is where you have files with illegal characters, lower-case file names, missing extensions, and other assorted weirdness. It depends on file extensions to tell it what type of file it's dealing with. You may recall that a file with no extension is treated as a *.DO* file—even if it isn't! At best, you can expect *LAPDOS.BA* to bomb (no big deal), and at worst, you can get a cold start (a tad more serious).

By the way, if *LAPDOS.BA* crashes, just press *F8* on the notebook to get the main menu; then rerun *LAPDOS.BA*. Provided you haven't done anything on the PDD window in the meantime, you need only do the *Log* command on drive P: to get back on track.

Which brings up another point. According to the manual, you cannot *Log* on drive P: when using the notebook. Tain't so. You can exit *LAPDOS.BA*, do whatever you like, and then restart *LAPDOS.BA*, *Log* drive P:, and your changes will be

updated on the PDD window. The only stipulation is that you don't attempt to perform any operations in the PDD window until you've got *LAPDOS.BA* running and have logged drive P: back in.

Owners of PG Design RAM banks beware. If you're using the *MENU.BA* program, it must not be in the invisible mode, where it's named *0MENU*, or you'll have problems. Leave it named *MENU.BA* or any legal file name.

Though notebook-to-PC copies filter out illegal characters in MS-DOS file names, copies in the other direction (to the notebook) don't have as much protection. When I tried to copy a PC file named *123456.BA* to the notebook, *LAPDOS.BA* crashed.

OTHER NOTES:

The manual doesn't mention that the *Format* command will format the notebook. Clearing it of all files except *LAPDOS.BA*. You could use that as a quick way to change environments—copy all the files to the PC, "format" the notebook, and then copy in the new ones you want.

The *README* file states that the background RAM resident mode of operation is no longer available. Too many other TSR-type programs can cause problems with each other, so Traveling Software thought it unwise to risk your valuable files in such an environment.

Remember, *LAPDOS.BA* must be run before *LAP.EXE*, or the PDD won't log. If that happens, you must *Quit LapDOS II* and restart it after running *LAPDOS.BA*.

I would randomly get Divide Overflow Errors when starting *LapDOS II*, aborting back to the DOS prompt. Usually, if I just ran it again it would work. When I switched to the 1400LT's slow processor speed, it would never fail. Switching back to fast speed brought the problem back. Traveling Software was unable to determine the cause and hadn't heard of this problem on any other 1400LT's.

The spiral bound manual is very thorough and well written. It proceeds in a logical order and seems to cover everything. It includes a sec-

tion telling not only what to put in an *AUTOEXEC.BAT* file but how to make the *AUTOEXEC.BAT* file. Appendices cover problems and error messages, creating and modifying an *AUTOEXEC.BAT* file, and setting the baud rate for different drives. It has a section on exchanging data between *T-base* and other programs.

SUMMARY

Even with a few flaws, most relatively minor, *LapDOS II* is the best thing I've seen for transferring files from machine to machine, machine to disk, and format to format. Its power, simplicity, and ease of use make *LapDOS II* a valuable tool for anyone who transfers files on a regular basis or transfers large numbers of files. The price is quite reasonable for the service it performs, and the upgrade policy for *LapDOS* owners makes it a bargain. For me, it's indispensable.

□

Manufacturer's Specifications:

LapDOS II: \$129.95
LapDOS (upgrade): \$49.95

Traveling Software
 18702 North Creek Parkway
 Bothell, WA 98011
 (800)343-8080
 (206)483-8088 (in Washington)

System requirements:

An IBM PC-, AT-, or XT-compatible computer with an RS-232C serial port
 Tandy PDD-1, PDD-2, Brother FB100, or Purple Computing disk drives, or
 A Tandy Model 100/102, 200, or NEC 8201A notebook computer
 A version of MS-DOS or PC-DOS 2.0 or higher
 66,000 bytes of free memory in PC's
 17,000 bytes of free memory in notebook computers

LapDOS cable and cable converter included.

Ending Your Computer's Solitude

Making an easy and cheap null modem cable.

by David O. Rowell

If you have two computers, or have a friend with a computer, you may have tried to get the two computers to "talk" to each other directly, by connecting them with cables. Probably, you connected them through the serial ports, using a gizmo called a "null modem adapter" and an RS-232 serial cable, or a combination of the two called a "null modem cable." Anyone reasonably proficient with a low-wattage soldering iron can make a serviceable null modem cable in an hour for less than ten dollars.

A null modem is nothing more than a device to take care of some connections and protocol details that a modem or other well-behaved communications device would normally provide. A null modem is required to get two computers communicating mainly because you're trying to get two devices—normally designed to talk all the time—to listen to each other. By design the computer's serial port sends information to a serial printer or modem through one wire, often getting information back from the *peripheral* device on another wire. On the other hand, also by design, a printer or modem receives information, sometimes sending information back to a computer. So you have two different kinds of devices: one (the computer) that's supposed to send, one (the peripheral) that's supposed to receive. What you are trying to do with

a null modem cable is to trick your talkers into the listening role.

This turns out to be very simple. Just swap the talk and listen wires somewhere in the cable. The communication protocols, the way the computers set the *rules* of communication, become a conversation where the computer says in effect, "Are you ready to receive?" and the other device says, "Yes, I am." or "No, wait a bit." Most *terminal* programs (pro-

*This turns out to be
very simple.
Just swap the talk
and listen wires
in the cable.*

grams that turn your computer into a *terminal*, or peripheral, for another computer, allowing them to communicate) are not prepared to handle this "handshaking" (*Telcom* certainly isn't) so you can take care of this in the null modem cable, too.

THE BIRDS, THE BEES,
AND THE DB-25 CONNECTOR
The serial port on the Models 100/

102/200, and most other computers, is a 25-pin, 2-row connector with 12 connections in one row and 13 in the other. The shell around the connections is vaguely "D" shaped so it's called a 25-pin D connector, DB-25. DB-25 connections come in two varieties, male and female. The RS-232 port on the Model 100 has holes—it's female. To connect to it, you need a male DB-25 connector, with pins. Now, check the RS-232 port on the

Parts Specifications:

Here's a list of the Radio Shack part numbers:

Solder type 25-pin male D connector: 276-1547, \$1.99
Solder type 25-pin female D connector: 276-1548, \$2.99
Hood for these connectors: 276-1549, \$1.99
Solder type 9-pin male D connector: 276-1537, \$1.49
Solder type 9-pin male D connector: 276-1538, \$2.49
Hood for these connectors: 276-1539, \$1.99
Bulk 4-conductor phone wire: 278-1320, \$0.10/foot

Make the cable a little longer than you think you need, but not longer than 25 feet or so because you may have interference problems.

Pin on Model 100	connect to	Pin on Other Computer
1 - GND, ground		Pin 1 - GND
2 - TR, transmit data		Pin 3 - RD
3 - RD, receive data		Pin 2 - TD
7 - GND, signal ground		Pin 7 - GND

On Each Connector Jumper These Two Pins:

- 4 - RTS, request to send
- 5 - CTS, clear to send

On Each Connector Jumper These Three Pins:

- 6 - DSR, data set ready
- 8 - CD, carrier detect
- 20 - DTR, data terminal ready

Figure 1. The pins used in a null modem and how you connect them on 25-pin, D connectors. Don't worry about the rest of the pins.

other computer to tell whether it's male or female. Many newer computers have a 9-pin D connector for the RS-232 port. Check your computers' documentation for the pin assignments corresponding to those signals listed in the chart in Figure 1 and make the hook-ups accordingly.

To make a null modem cable you'll need a flexible four-wire cable, the appropriate connector for the two computers, and hoods to cover the connections and provide a grip for taking off and putting on the connectors easily. The cable could be as simple as a piece of four-wire telephone cable or as fancy as some of the industrial cables (mostly in shielding and insulation) you'll see at stores. Naturally, the phone cable needs much more careful handling than the industrial cable. No significant electrical current flows through the wires, so mechanical strength really determines the wire size. You can get the parts from many electronics stores, but Radio Shack Computer Centers and stores should carry the parts you need (see the Parts Specifications box).

PUTTING THEM TOGETHER

When putting together the cable, swap one computer's transmit data

line with the other's receive data line, and they can communicate. The DB-25 connectors' pins are numbered on the solder side to assist pin identification. Look closely, but even with normal vision you may need a magnifier to see them. *Jumper* (connect with a short wire) pins 4 and 5 on both connectors, and also jumper pins 6, 8 and 20 on both connectors. These jumpers let a computer ask, "Are you ready?" and will loop back so it can answer its own question with a "Yes."

Put both hoods over the connectors *before* hooking them to the computers, and you'll be sure not to forget to put them on. Keep the connectors clean and you'll make much

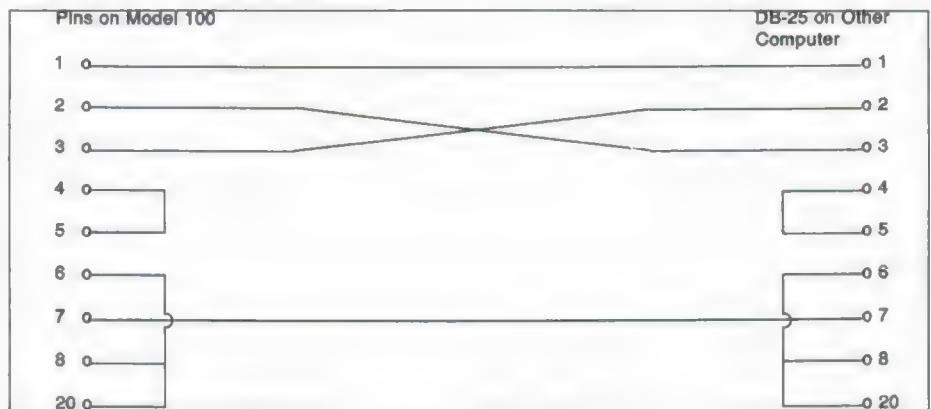


Figure 2: Diagram of the connections in the null modem cable.

The cable could be as simple as a piece of four-wire telephone cable or as fancy as some of the industrial cables.

better connections. Remember that you're dealing with low voltages and currents, so the solder connections must be good. When making up the cable be careful not to bridge between pins. Don't strip any more insulation from a wire than you need to fit the pin, or you may develop a short circuit in the future. You can make the jumper between pins 4 and 5 with a short bare wire. The jumper between 6, 8 and 20 must be carefully insulated since it goes right over pin 7, the signal ground connection. If a solder connection doesn't look perfect—it isn't. Wrap the cable with lots of tape where it goes through the hood to give a good grip and some strain relief.

Next Month:

Using a null modem to connect your Tandy notebook computer to a desktop computer.

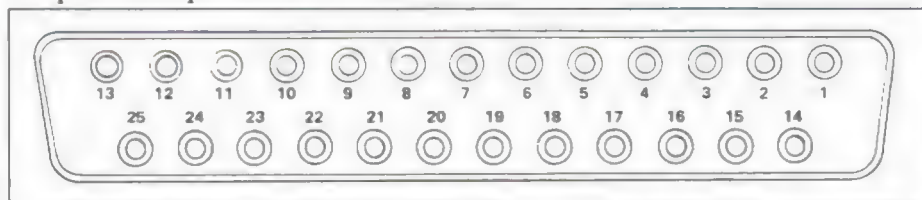


Figure 3: Numbering on a female RS-232 connector. (see Model 100 manual)

COMPATIBILITY: Tandy 100/102 only

HOTKEY: Escape Artist

Burn up your keyboard with simple, two-key commands.

by Mike Nugent

HOTKEY is a little utility that performs some special magic with simple, two-key commands. Whether you're in BASIC, TEXT, ADDRSS, SCHEDL, or Detroit, this mini-Houdini can:

- * Display the current day, date, and time
- * Automatically type the day, date, or time
- * Send a form feed to your printer
- * Offer a choice of five-space or eight-space tabs
- * Turn itself off

It can be customized, uses only 379 bytes of RAM, and as a bonus, even fixes the Model 100 "date bug." Sound good? Then on with the show!

*It can be customized,
uses only 379 bytes of RAM,
and even fixes the Model 100
"date bug."*

First, type the program *exactly* as listed, except that remark lines need contain only the line number and the apostrophe (or REM). Double check for errors; then save it to RAM as HOTKEY.BA. And save it to tape or disk as well, so you'll never have to type it in again.

CAUTION: Because HOTKEY uses machine language, mistakes here can vaporize your files. Before proceeding, backup anything important! Nuff said.

INSTALLING HOTKEY

To install HOTKEY, first kill all other .BA programs except PG Design's MENU.BA, Micro Demon's SUPERA, and

Tri-Mike's DVORAK, if you use any of those. Note that some BASIC programs don't show a .BA extension (for example, Traveling Software's UR-2, RANDOM, TWORD+). When in doubt about a file, kill it. Now run HOTKEY.BA. It installs in less than a minute, shrinking itself to 379 bytes, then returns to the main menu, ready to use. If you get a Load Error message, see the LOAD ERROR section below. **Never edit the installed HOTKEY.BA!** You can list it (it will look weird), but don't edit it!

USING HOTKEY

To turn HOTKEY on, just run it. The menu quickly returns, and HOTKEY.BA disappears. Why? So you can't kill it. Killing it while it's active can cause a cold start. It

```

10 REM-----1-----2-----3-----
   ---4-----5-----6-----7-----
   ---8-----9-----012345678
20 REM-----1-----2-----3-----
   ---4-----5-----6-----7-----
   ---8-----9-----0-----1-----
   ---2-----3-----4-----5-----
   ---6-----7-----8-----9-----
   ---0-----1-----2123456
30 PRINT "HOTKEY.BA": PRINT "Copyright 1
987 Tri-Mike Network East"
40 PRINT "by MJ Nugent, Jr.": PRINT "All
rights reserved"
50 PRINT "Installing...."
60 '--- m/l sizes, variables ---
70 READ S1, S2, DFLAG, DIR, YO, YT, YR
80 '--- locate HOTKEY in RAM ---
90 PRG=PEEK(63100)+256*PEEK(63101)
100 '--- set line 10 & 20 pointers ---
110 L10=PRG+5
120 P20=PEEK(PRG)+256*PEEK(PRG+1)
130 L20=P20+5

```

Continued.

A self-modifying BASIC program that gives you simple, two-keystroke control of your Tandy 100/102.

FOUR MODEL 100 BOOKS from Short, Simple And Basic

The Model 100 Program Book

by Terry Kepner and David Huntress

This book contains 51 useful and interesting BASIC programs designed for home, office, and educational uses. Program examples include: Bar Graph, Depreciation, Annuity, Pie Chart, Forms creation, Invaders game, Memory scan, Touch typing tutor and many others.

Only \$12.95 plus shipping.

60 Business Applications Programs for the TRS-80 Model 100 Computer

by Terry Kepner and Mark Robinson

Here are 60 powerful programs for interest calculations, annuities, depreciation, invoices, breakeven sales analysis, and more.

Only \$17.95 plus shipping.

Inside the Model 100

by Carl Oppedahl

"...an excellent Guide" —*New York Times*

Inside the TRS-80 Model 100 is a thorough guide to the internal workings of the Tandy Model 100, the best selling portable laptop computer. This book is a valuable source of information for those who wish to learn assembly language programming on the Model 100. Other areas include: Disassembled ROM routines; Keyboard scanning; UART, RS-232C, and modem; Clock/calendar chip; Interrupt handling; 8085 instruction set.

Only \$19.95 plus shipping.

User Guide and Applications for the TRS-80 Model 100 Portable Computer

by Steven Schwartz

Gain expertise on the Tandy Model 100 with 14 ready-to-run programs for business. This package includes programs for statistics, graphics, sound, and more. With cassette tape.

Only \$42.95 plus shipping.

Or buy them separately—the book is only \$17.95; the cassette tape is only \$25.00, plus shipping.

Please add \$2.00 shipping and handling for each book ordered.

Portable Computing International Corp.

P.O. Box 428

Peterborough, NH 03458-0428

Your source for Tandy Model 100/102 books.

PROGRAMMING SIDEBAR

DOS Patches!

Indispensable Tandy Portable Disk Drive software modifications.

Tandy's FLOPPY.CO. is fatal to any .BA program containing machine language, including SUPERA, MENU, DVORAK, and now HOTKEY. Fortunately, you can patch FLOPPY to prevent it from moving .BA programs:

PDD-1 (100K disk drive) users type:

```
CLEAR 256,59400
and press ENTER
LOADM "FLOPPY.CO"
and press ENTER
POKE 61571,154
and press ENTER
POKE 61572,249
and press ENTER
POKE 61573,0
and press ENTER
SAVEM "NEWDOS.CO", 59400,
62959, 59400
```

and press ENTER

The patched FLOPPY is renamed NEWDOS.CO to indicate its patched status. Save NEWDOS to tape or disk, and from now on, always use NEWDOS instead of FLOPPY. To save it to tape, type: CSAVEM "NEWDOS", 59400, 62959, 59400 and press ENTER

With NEWDOS on tape, you needn't ever re-IPL using Tandy's IPL.BA program. Just load NEWDOS from tape:

```
CLEAR 256,59400
and press ENTER
CLOADM "NEWDOS"
and press ENTER
SAVEM "NEWDOS", 59400,
62959, 59400
and press ENTER
```

If you have NEWDOS on disk and ever need to re-IPL using IPL.BA, then immediately after doing so, load NEWDOS from disk and kill FLOPPY.

PDD-2 (200K disk drive) users type:

```
POKE 62116,154
and press ENTER
POKE 62117,249
and press ENTER
POKE 62118,0
and press ENTER
POKE 60001,91
and press ENTER
```

Because of the design of FLOPPY for the PDD-2, these changes can't be saved to tape or disk. Make them each time you install FLOPPY from disk.

will reappear when you turn it off.

HOTKEY commands work anywhere except **TELCOM's** *Term* mode and the main menu. To execute a command, press **ESC** followed by the appropriate key:

Hotkey Meaning

ESC-V	(View)—Display day/date/time on top line until any key is pressed
ESC-D	(Date)—Type date in MM/DD/YY format
ESC-T	(Time)—Type time in HH:MM:SS format
ESC-W	(Week)—Type day of the week, e.g., "Mon"
ESC-F	(Form)—Send form feed to printer
ESC-O	(Off)—Turn HOTKEY off
ESC-TAB	(Tab)—Five-space tab

The Date, Time, and Week commands actually *type* the information as though it came from the keyboard. The View command only *displays* it on your screen. The Form command sends a form feed, **CHR\$(12)**, to the printer or beeps if the printer isn't ready. This is handy for printing multiple files to a buffered printer. Pressing **ESC-F** after each print command ensures that your files are separated by form feeds; no need to wait until one report has printed before printing the next one. Whereas the normal **TAB** types a tab character, **CHR\$(9)**, **ESC-TAB** types the necessary number of spaces to produce a five-space tab. The Off command turns **HOTKEY** off, beeps to acknowledge the command, and makes **HOTKEY.BA** visible again on the main menu.

EXTRAS

Have you met the notorious Model 100 *date bug*, the one where, for no apparent reason, it's suddenly next year? **HOTKEY** ends this game of *musical years* and keeps you solidly planted in 1988 (or whatever year you're actually living in). Each year, you can change the last data item in line 580 to match the current year.

You can customize some features of **HOTKEY**. If you prefer date and time in the shorter MM/DD and HH:MM format, change the 9 in line 840 to a 6.

To change the printer character the Form command sends, replace the 12 in line 880 with another value. A value of 15, for example, puts some printers into condensed mode.

Though I don't advise it, changing "238, 8" in line 600 to "127, 127" leaves **HOTKEY** visible when active. It will appear on the menu as **+OTKEY.BA**, the + being an extra safeguard against its accidental demise. But many DOS's and utilities can easily zap it and blow your machine off into the weeds.

LOAD ERROR

Packing machine code into a .BA program (as **HOTKEY** does) can be a complex affair, because you can't predict where the program will reside in memory. Since the values of certain bytes depend on the program's location, sometimes a byte contains a value **BASIC** misinterprets. The *Load Error* message warns you of such a byte. You must move the code higher in memory, a byte at a time,

```

140 '--- install m/l in line 10 ---
150 BASE=L10: N=S1-1: GOSUB 460
160 '--- skip junk betw. lines ---
170 FOR I=1 TO 6: READ A: NEXT
180 '--- install m/l in line 20 ---
190 BASE=L20: N=S2-1: GOSUB 460
200 '--- find HOTKEY dir entry ---
210 A$="": FOR I=3 TO 10: A$=A$+CHR$(PEEK(DIR+I)): NEXT
220 IF A$<>"HOTKEYBA" OR PEEK(DIR)<>128
  THEN DIR=DIR+11: GOTO 210
230 '--- store it in m/l code ---
240 MSB=INT(DIR/256): LSB=DIR-MSB*256
250 POKE L10+DFLAG,LSB: POKE L10+DFLAG+1,MSB
260 '--- store yr for date$ fix ---
270 YR$=STR$(YR)
280 POKE L10+YT,ASC(MID$(YR$,2,1))
290 POKE L10+YO,ASC(RIGHT$(YR$,1))
300 '--- setup to kill these lines ---
310 OPEN "KILLER.DO" FOR OUTPUT AS 1
320 FOR I=30 TO 1090 STEP 10
330   PRINT#1, STR$(I)
340 NEXT
350 CLOSE
360 '--- stuff cmds into keyboard buffer ---
370 A$="KILL"+CHR$(34)+"KILLER.DO"+CHR$(13)+
  "MENU"+CHR$(13)
380 FOR I=1 TO LEN(A$)
390   POKE65449+2*I,ASC(MID$(A$,I,1))
400   POKE65450+2*I,0
410 NEXT
420 POKE 65450,I-1
430 '--- waste lines, end install ---
440 MERGE "KILLER.DO"
450 '--- install/relocate m/l ---
460 FOR I=0 TO N
470   PTR=BASE+I
480   READ A: IF A=>0 THEN 540
490   READ A, B: REL=A+256*B
500   FX=REL+L10
510   B=INT(FX/256): A=FX-B*256
520   IF A=0 OR B=0 OR A=13 OR B=13 THEN
     BEEP: PRINT "Load Error!": END
530   POKE PTR+1,B: I=I+1
540   POKE PTR,A
550 NEXT
560 RETURN
570 '--- s1, s2, dflag, dir, yo, yt, yr ---
580 DATA 108, 226, 1, 63842, 36, 29, 88
590 '--- line 10 m/l code ---
600 DATA 33, 255, 255, 126, 238, 8, 119, 40
610 DATA 3, 26, 238, 99, 18, 33,-1, 28
620 DATA 0, 235, 42, 222, 250, 34,-1, 14
630 DATA 0, 235, 34, 222, 250, 201, 62, 56
640 DATA 214, 48, 50, 46, 249, 62, 55, 214
650 DATA 48, 50, 45, 249, 42, 82, 246, 17
660 DATA 239, 84, 223, 202,-1, 59, 0, 17
670 DATA 6, 89, 223, 194,-1, 65, 0, 62

```

Continued.

until all the bytes are acceptable to BASIC. Here's how:

1. Type **NEW** and press **ENTER**. Type **KILL "HOTKEY.BA"** and press **ENTER**.
2. Load a fresh copy of **HOTKEY.BA** from disk or cassette, and save it to RAM as **HOTKEY.BA**.
3. Edit line 10. Insert an **X** right after the **REM**, making it **REMX**.
4. Edit line 110. Change the **+5** to **+6**.
5. Edit line 2000. Change the **+5** to **+6**.
6. You've moved the program up one byte in memory.

Try running it again. If it still fails, repeat steps 1-6, inserting another **X** (for example, **REMX**) and adding 1 more to the numbers in lines 110 and 2000. Keep repeating these steps until it works. One or two **X**'s usually does the trick. If you've added 5 or 6 **X**'s, I suggest you carefully double check your program listing for errors.

COMPATIBILITY

Compatibility with other programs depends on how they handle a system vector that **HOTKEY** uses:

- 1) **TS-DOS** likes it just fine.
- 2) There's no conflict with **MENU.BA**.
- 3) It's completely compatible with **DVORAK** (of course!).
- 4) **SUPERA** turns **HOTKEY** off, and vice versa, so use them one at a time.
- 5) **Ultimate ROM II** cooperates with a few exceptions:
 - a) **UR-2** programs exit to the main menu rather than to **UR-2**'s menu.
 - b) After using **HOTKEY**'s View command with **UR-2**'s **VIEW80** activated, the screen clears completely. Scrolling up or down restores the missing lines.
 - c) **CTRL-J**, **CTRL-O**, and **CTRL-L** functions of **UR-2**'s **TEXT** are disabled with **HOTKEY** active. To reenable them, turn **HOTKEY** off via its **ESC-O** command.
 - d) After exiting a text file from **UR-2**, the main menu shows **TEXT** as **TEXT.** (with an added period). These problems appear to be harmless so far, but be cautious. With any new software, make frequent backups until you know what works with what.

FUTURE POSSIBILITIES

I meant **HOTKEY** to be a small, simple utility. But with all those other keys on the keyboard just begging to become part of **HOTKEY**'s repertoire, I wasn't sure when to stop adding commands. I can easily imagine **HOTKEY** growing larger and more powerful in the future. If you'd care to share your ideas and comments, please contact me care of this magazine or directly via CompuServe ID 71426,1201. Or just drop by Tri-Mike East—and bring a pizza!

Biography. Ummm.... drat! Holy cow lemme think.... Oh, yeah: Mike Nugent is the president of Tri-Mike Network East and the developer of the Dvorak Keyboard System. He once had a parakeet.

```

680 DATA 1, 50, -1, 66, 0, 201, 62, 1
690 DATA 254, 2, 250, -1, 94, 0, 202, -1
700 DATA 114, 0, 62, 1, 61, 202, -1, 94
710 DATA 0, 50, -1, 76, 0, 33, 255, 255
720 DATA 126, 35, 34, -1, 85, 0, 225, 20
1
730 DATA 205, 214, 18, 245, 254, 27, 19
4, -1
740 DATA 63, 1, 62, 2, 195, -1, 65, 1
750 '--- junk between lines ---
760 DATA 0, 0, 0, 0, 0, 0
770 '--- line 20 m/l code ---
780 DATA 205, 214
790 DATA 18, 245, 205, 233, 15, 254, 27
, 202
800 DATA -1, 63, 1, 33, -1, 71, 1, 34
810 DATA -1, 85, 0, 254, 84, 194, -1, 143
820 DATA 0, 205, 15, 25, 195, -1, 151, 0
830 DATA 254, 68, 194, -1, 156, 0, 205,
47
840 DATA 25, 62, 9, 195, -1, 51, 1, 254
850 DATA 87, 194, -1, 169, 0, 205, 98, 2
5
860 DATA 62, 4, 195, -1, 51, 1, 254, 70
870 DATA 194, -1, 193, 0, 219, 187, 230,
6
880 DATA 238, 2, 62, 12, 245, 196, 98,
118
890 DATA 241, 204, 63, 109, 195, -1, 20,
1
900 DATA 254, 86, 194, -1, 9, 1, 42, 224
, 250, 229, 33, 243, 127, 34, 224, 250
910 DATA 33, 1, 254, 43, 229, 17, 70, 2
55, 6, 40
920 DATA 229, 213, 197, 205, 66, 37, 42
, 57
930 DATA 246, 229, 33, 1, 1, 205, 124,
66
940 DATA 205, 93, 66, 225, 34, 57, 246,
205
950 DATA 112, 93, 205, 214, 18, 193, 20
9, 225
960 DATA 235, 205, 66, 37, 225, 205, 34
, 70, 225, 34, 224, 250
970 DATA 195, -1, 20, 1, 254, 79, 194,
-1
980 DATA 25, 1, 205, -1, 0, 0, 205, 98
990 DATA 118, 62, 1, 195, -1, 56, 1, 25
4
1000 DATA 9, 194, -1, 63, 1, 33, -1, 79
1010 DATA 1, 34, -1, 85, 0, 58, 58, 246
1020 DATA 61, 6, 5, 144, 242, -1, 42, 1
1030 DATA 128, 79, 120, 145, 60, 50, -1
, 76
1040 DATA 0, 62, 3, 50, -1, 66, 0, 241
1050 DATA 175, 225, 201, 62, 1, 50, -1,
66
1060 DATA 0, 241, 225, 201, 42, 98, 117
, 102
1070 DATA 102, 101, 114, 42, 32, 32, 32
, 32
1080 DATA 32
1090 '--- turn HOTKEY on/off ---
2000 CALL PEEK(63100)+256*PEEK(63101)+5:
MENU

```

End of listing.

DEFUSR appears monthly to answer your questions about Tandy notebook computers.

*Send your queries to: DEFUSR, PORTABLE 100,
P.O. Box 428, Peterborough, NH 03458-0428.
Please enclose a stamped, self-addressed envelope for our reply.*

QUESTIONS FROM A NEW MODEL 200 OWNER

I have recently purchased a Tandy Model 200 Computer. I am extremely pleased with it, but I have some questions that I hope you can take the time to answer.

I chose the Model 200 over the Model 100 solely because of the larger display. I bought it primarily to use as the computer portion of a portable amateur radio packet station. Packet radio is a mode of communication which allows a computer to be connected to a radio through a device called a terminal node controller. The operator runs a regular telephone modem program, and anything entered from the keyboard is sent to the TNC (which looks like a modem to the terminal program), and the TNC formats the messages into packets, activates the radio transmitter, and send the messages over the air. Any kind of data which the terminal program can handle may be sent over the air. Disk files may be transferred from one packet station to another, and there are packet radio bulletin boards that operate much like telephone bulletin boards. With one exception, the TELCOM program of the Model 200 worked very well with packet. The one problem is that the TELCOM program will transfer only text data. Is a program available that uses the RS-232 port and will transfer any kind of data?

Another problem I have is with the documentation. The documentation included with the Model 200 is pathetically lacking in information useful to anyone who wants to do more than run the built-in programs. For example, if you look up the opening statement in the BASIC manual you

*There seems to be an
incredible lack
of information
about the Model 200.*

read that you can open RAM files, cassette files, modem files, and RS-232 files. Yet there is no example of an open statement for an RS-232 file. You have to look at a sample program in the back of the book to see how this is done.

Is there any way for a BASIC program to check or change the status of the RS-232 lines? The technical manual explains how to make calls to ROM routines to do this, and the results of the operations are placed in CPU registers, but how would a BASIC program get this data? Would

you have to write a machine language program to copy the contents of the registers into memory so a basic program could get to them?

My last questions have to do with compatibility of hardware and software between the Models 100/102 and the 200. Can I assume that a product (either hardware or software) will not work with the Model 200 if the advertisement does not specifically state that it will? Would it be possible to publish in your magazine a chart of products showing which ones work with which computers? People would probably ask why I didn't get a Model 100 since it seems to be supported better than the Model 200, but I was able to get my Model 200 on sale for \$350 while Model 100s were still \$499, and I needed the larger screen anyway.

There seems to be an incredible lack of information about the Model 200. Or maybe I just haven't been looking in the right places. If you know of any sources which may be of some use please let me know. Even the Tandy retail and computer store personnel are remarkably uninformed. I really like the computer, but if I can't find hardware or software to do some of the things I want to do it's going to be of limited usefulness. Perhaps if you could publish this letter and your reply in the I/O column of your magazine it would benefit other Model 200 owners. Thank you for taking the time to read

this letter, and I look forward to seeing your reply.

**Michael A. Newell
Toney, AL**

The XMODEM for the T200 will transfer the files on the RS-232 port, but it requires an XMODEM program on the other end to decipher the data. I need a bit more information about what you are sending and what happens at the other end.

I suggest you get David Lien's book, The Tandy 200 Portable Computer. It is a complete training manual on the T200, especially BASIC (but a little skimpy on MSPLAN).

Changing the TELCOM status is easy. Put this line into your BASIC program OPEN"COM:37E1E"FOR OUTPUT AS 1 to open the RS-232 port for transmitting data at 300 baud(3), seven bit data words (7), even parity (E), one stop bit (1), and XON/XOFF protocol. There is no way for BASIC to query the RS-232 port to check its current setting (if you knew the address that TELCOM uses to query STAT, you could use that with a CALL command to retrieve the information).

For commercial software/hardware, you're correct, unless it specifically says it's for the 200, it isn't. As for software, BASIC programs (those that do not use CALL's) will work just fine. Machine code won't work without modifications by a skilled programmer.

And keep watching these pages for more information on the Tandy 200.

-TK

RESTORING FUNCTION KEYS

I am wondering if you could advise me what the two CALL lines are that would restore the F1-F8 function keys on the Tandy 200 to their normal functions. For example, in your April 1988 issue Listing 1 of Mike Aiello's SHELL.BA program (p. 12), there are the following two lines: 32260 CALL 23164,0,23366 and 32265 CALL 27795

These two lines restore the function keys on the Model 100 to normal. What I need are the two similar call lines that will restore function keys to normal on the Tandy 200.

**Kenneth G. Oxley
Dayton, OH**

These should do it: CALL 28192, 0, 28357 and CALL 33747

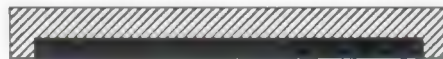
-MN

A MODEL 100 FORTH?

Thank you for your valuable magazine. I would like to know if one can get a FORTH package for the Model 100/102. I would like to program in FORTH on this machine. Where can I get it?

**Iver Iversen
Jacksonville Beach, FL**

You can get a copy of FORTH for the Tandy 100 right off our Portable BBS! Just call 603-924-9770 with your computer, make up a personal password. Answer linefeeds yes, nulls zero, and you can go straight to the download menu



*You can get a copy of
FORTH
for the Tandy 100
right off our
Portable BBS!*



where you'll find a list of the files available for downloading (the odd choices in the menu, the even ones are for actually downloading the files).

-TK

ANOTHER RIBBON TO RE-INK?

Does anybody know of a method for re-inking DMP-100 ("LINE PRINTER VII") ribbons? My dad, who is on Social Security, didn't get started "computing" until after he retired. He uses a DMP-100. I'll pass the information about re-inking on to him, but the DMP-100 ribbon is quite different....

**John S. Neufeldt
Tucson, AZ**

Sorry, but we can't help you there. Does anyone else have an answer?

-TK

CTS AND DSR WOES

I am the owner of a RS Model 100 computer and need assistance in connecting the computer's RS-232C interface with external equipment. I have exhausted my sources of help

and would appreciate any assistance that you can offer.

The equipment that I am attempting to interface with has the capability for returning CTS and/or DSR signals back to the Model 100's RS-232C interface for controlling transmitted data from the 100, but the Model 100 (using TELCOM) does not respond to these inputs. I mistakenly believed from the information listed by my Radio Shack owner's manual that the Model 100 could interface with and be controlled by external equipment via the RS-232C port control lines CTS and DSR.

I need your support by supplying me with a program (or a recommendation for a source of software or other means) that will enable the Model 100 to accept and be controlled by the CTS and/or DSR signals on the RS-232C interface while uploading data in TELCOM.

I need to mention that the Xon/Xoff protocol will stop or start transmitted data during upload, but this capability will not work in many of my applications, since the equipment I am interfacing with does not use this protocol.

Any help from you will be greatly appreciated.

**John E. Riggi
New York, NY**

As you mentioned, the Radio Shack manual is wrong. The Model 100 provides for these signals in hardware, but TELCOM ignores them. There are a couple of ways to solve the problem, the easiest being a cheap and simple hardware modification described in "The Art of Hardware Handshaking" (Portable 100, Dec. '87). Back issues are available. The other way is to forego TELCOM and write your own communications software, controlling the signals directly via INs and OUTs to ports. The ports and a great deal of other information are covered in Carl Oppedahl's book, Inside the TRS-80 Model 100. For ordering information, see the advertisement on page 22 of this issue.

-MN



Cross-Bank BASIC I/O

It's hard to believe that the Tandy 200 is three and a half years old. I got mine when they were first released, and I have been waiting three and a half years for someone to develop the code for inter-bank programming. It turned out to be a do-it-yourself project for me, and I am happy to share it with *Portable 100* readers.

Unfortunately the Tandy 200 was not as popular as Tandy had hoped, and they are now "officially" not manufacturing them anymore. That's really too bad because the material I will present in this month's column will alter your perception of the Tandy 200, forever.

Last month's column about *PASTE.200* was for those who use *TEXT* and wanted to *CUT* and *PASTE* from one bank to another. This month is directed to the *BASIC* programmer who would like to access another bank for data storage and retrieval. A *BASIC* program running in one bank could use the full 19.5K RAM in each of the two remaining banks, for a really *LARGE* database.

XBASIC.CO (see Listing 1) is under 500 bytes and runs in the alternate LCD buffer, so it will not overwrite DOS or other *.CO* files being used. Once the *BASIC* program has executed the *LOADM"XBASIC"* command, all the routines can be accessed with the following call: *CALL 63600, command [,VARPTR(X\$)]* ([]=optional parameter).

In place of the word "command," above, you would type one of the

following integers:

Command	Meaning
0	CLOSE FILE
1	OPEN FOR INPUT
2	OPEN FOR APPEND
3	READ X\$
4	WRITE X\$
5	CHAIN FILE
6	DISPLAY FILES

0	CLOSE FILE
1	OPEN FOR INPUT
2	OPEN FOR APPEND
3	READ X\$
4	WRITE X\$
5	CHAIN FILE
6	DISPLAY FILES

XBASIC does internal error checking and will beep if there is a problem. *XBASIC* will not execute the wrong instruction and offers the *BASIC* programmer a result descriptor byte so the *BASIC* program can

*You've always
wanted to. Now
you can!*

trap and handle problems.

The result descriptor (RD) byte is at 63605, and can be *PEEK*ed either before or after a *CALL* to *XBASIC*. There is also an address pointer for the opened file at 63603 and 63604, in the LSD/MSD format.

THE COMMANDS

CLOSE FILE: *CALL 63600,0*. This command sets RD to equal 0 and you can *CALL* it at anytime. This also sets the address pointer to 0.

OPEN READ: *F\$="2:file":CALL*

63600,1,VARPTR(F\$). This command first checks the RD, and if RD equals 0, then it is OK to *OPEN*. The *OPEN READ* command sets RD equal to 1 and sets an address pointer to the beginning of *FILE.DO* in bank number 2.

If *RD<>0*, then it is NOT OK to *OPEN* (because it's already open). A *BEEP* is sent and the open file is closed (RD set to 0).

OPEN WRITE: *F\$="2:file":CALL 63600,2,VARPTR(F\$)*. This command first checks the RD, and if RD equals 0, then it is OK to *OPEN*. The *OPEN WRITE* command sets RD equal to 2 and sets an address pointer to the end of *FILE.DO* in bank 2.

If *RD<>0*, then it is NOT OK to *OPEN* (again, because it's already open). A *BEEP* is sent and the open file is closed (RD set to 0).

READ: *X\$="":CALL 63600,3,VARPTR(X\$)*. This command first checks the RD, and if RD equals 1, then it is OK to *READ*. The *READ* command leaves RD equal to 1 unless an end of file—*CHR\$(26)*—is found. If an end of file is found, then RD is set to 17. When *XBASIC* returns from this *CALL*, *X\$* is set to the data that was read. The data read is from the address indicated by the pointer to the next carriage return, to an end-of-file character, or 255 bytes, whichever comes first. The address pointer is moved appropriately to the end of the data just read from *FILE.DO* in bank 2.

If *RD<>1*, then it is NOT OK to *READ*. Perhaps end of file was encountered in the previous *READ*, or

maybe the file was opened for output. A BEEP is sent, and no read will take place. The RD is not altered, so the user can inspect the byte to see why a BEEP occurred.

WRITE: X\$="data":CALL 63600,4,VARPTR(X\$). This command will first check the RD, and if RD equals 2, then it is OK to WRITE. The WRITE command leaves RD equal to 2 unless an OM? condition is found in the destination bank. If an OM? is found then RD is set to 18 and the WRITE does not occur.

When XBASIC returns from this CALL, X\$ is appended to the data file that was opened for output. X\$ is written to the address indicated by the pointer (end of file). The address pointer is moved appropriately to the new end of the data file in bank 2. Carriage returns are not automatically added.

If RD<>2, then it is NOT OK to WRITE. Perhaps OM? was encountered in the previous WRITE, or maybe the file was opened for input. A BEEP is sent, and no write will take place. The RD is not altered so the user can inspect the byte to see why a BEEP occurred.

CHAIN: BNK=2:IPL"file.BA":CALL 63600,5,BNK. This command jumps to bank (BNK) and executes FILE.BA. This is the only command that does NOT want the bank to be part of the file name. The file indicated by IPL must have a file extension. You can CHAIN to a .DO file and that would be like switching banks and jumping into TEXT to edit the specified file.

FILES: BNK=2:CALL 63600,6,BNK. This command displays the

You can CHAIN to a .DO file

files in bank (BNK) 2.

These commands should NOT be used to address the current bank where the standard BASIC commands (OPEN, PRINT#, etc.) can be used. These commands offer new horizons for the Tandy 200 BASIC programmer.

SOME SAMPLE PROGRAMS

I've included a couple of sample programs to illustrate the usability of XBASIC:

REATST.BA (See Listing 2) is an example of using XBASIC to read from another bank. At the prompt, type in the file you wish to duplicate. That file will be copied to the current bank.

WRITST.BA (See Listing 3) is an example of using XBASIC to write to another bank. At the Client prompt, type in a name (use upper case only). WRITST writes the name to CLIENT.DO in either bank 1, 2, or 3. WRITST.BA determines which bank to write to, then opens the file for READING. It reads names, and when the proper position is found for the current entry, the RD byte is changed so that a write operation can be performed. This keeps CLIENT.DO in each bank sorted on data insertion.

The apt BASIC programmer could also develop RANDOM access to data files, as well as the sequential access we are now accustomed to. XBASIC.CO provides all the functions needed to create some pretty sophisticated applications.

by Paul Globman □

Editor's Note: We at Portable Computing were unable to test this program because we didn't have the necessary banks available.

```

1 *****
2 '* XBASIC.CO *
3 '* by Paul Globman *
4 '* Copyright (c) 1988 *
5 *****
10 FOR I = 63572 TO 64043
20 READ X:POKE I,X:SM=SM+X
30 NEXT
40 IF SM = 65761 THEN 60
50 PRINT"error in data":STOP
60 SAVEM"XBASIC.CO",63572,64043
1000 DATA 197,6,0,213,86,227,205,176
1010 DATA 155,209,235,193,35,19,11,121
1020 DATA 176,194,84,248,201,243,205,177
1030 DATA 155,251,122,201,195,118,248,0
1040 DATA 0,0,167,202,184,248,254,6
1050 DATA 202,230,249,254,5,202,172,249
1060 DATA 254,4,202,194,248,254,3,202
1070 DATA 98,249,71,58,117,248,167,194
1080 DATA 181,248,120,50,17,249,50,117
1090 DATA 248,205,198,249,35,35,95,205
1100 DATA 202,90,33,70,247,17,70,247
1110 DATA 1,6,0,205,84,248,195,229
1120 DATA 248,205,69,79,33,0,0,34
1130 DATA 115,248,34,116,248,201,50,17
1140 DATA 249,58,117,248,254,2,194,69
1150 DATA 79,229,126,254,0,111,38,0
1160 DATA 34,48,249,225,200,35,235,237
1170 DATA 34,75,249,42,115,248,34,45
1180 DATA 249,243,219,216,230,12,50,54
1190 DATA 249,56,0,235,34,58,249,33
1200 DATA 254,248,84,93,1,59,0,205
1210 DATA 84,248,62,0,211,216,71,58
1220 DATA 54,249,184,202,16,249,49,156
1230 DATA 249,205,71,76,62,0,254,4
1240 DATA 202,44,249,254,2,33,76,247
1250 DATA 54,68,35,54,79,245,205,124
1260 DATA 45,241,204,104,130,195,53,249
1270 DATA 33,0,0,1,255,255,205,168
1280 DATA 130,62,0,211,216,49,0,0
1290 DATA 245,58,17,249,254,4,194,82
1300 DATA 249,241,218,88,249,235,33,0
1310 DATA 0,205,84,248,235,245,241,34
1320 DATA 115,248,251,201,205,69,79,62
1330 DATA 18,50,117,248,251,201,58,117
1340 DATA 248,254,1,194,69,79,243,229
1350 DATA 58,255,248,71,14,1,17,112
1360 DATA 239,42,115,248,43,35,213,197
1370 DATA 205,177,155,122,193,209,254,26
1380 DATA 202,164,249,254,13,202,147,249
1390 DATA 18,19,12,194,121,249,43,35
1400 DATA 35,34,115,248,121,61,225,119
1410 DATA 35,235,33,112,239,217,251,201
1420 DATA 62,17,50,117,248,195,149,249
1430 DATA 243,125,205,213,249,33,221,244
1440 DATA 229,209,1,11,0,205,84,248
1450 DATA 205,26,28,58,86,248,79,195
1460 DATA 144,155,126,61,61,35,235,237
1470 DATA 245,126,214,48,205,213,249,241
1480 DATA 201,71,62,1,135,5,194,216
1490 DATA 249,230,12,50,255,248,50,86
1500 DATA 248,201,125,205,213,249,71,33
1510 DATA 181,242,205,105,248,254,255,20
0
1520 DATA 254,240,202,254,249,230,128,19
4
1530 DATA 5,250,17,11,0,25,195,238
1540 DATA 249,205,30,0,35,35,35,205
1550 DATA 105,248,231,62,8,61,202,35
1560 DATA 250,50,16,250,254,2,194,10
1570 DATA 250,62,46,231,195,10,250,62
1580 DATA 8,50,16,250,35,195,238,249
1590 REM END OF DATA End of listing.

```

Listing 1. The BASIC loader for XBASIC.CO.

Planning To Move?

Before you make the move, please let us know at least seven weeks in advance so you won't miss a single issue of Portable 100.

Simply clip out or photocopy this form, attach the old label, print your new address in the space provided, and mail it all to:

Portable
Address Change Dept.
P.O. Box 428
Peterborough, NH 03458-0428

**Attach label
 here, or write in
 old address**

NAME

ADDRESS

CITY

STATE

ZIP

EFFECTIVE DATE

```
0 REM REATST.BA
10 INPUT"x:file";F$
15 F$=LEFT$(F$+"",8)
20 F1$=RIGHT$(F$,6)
30 OPEN F1$ FOR OUTPUTAS1
40 LOADM"XBASIC":CALL63600,1,VARPTR(F$)
50 X$="":CALL63600,3,VARPTR(X$)
60 PRINT#1,X$;
70 IF PEEK(63605)=17 THEN MENU
75 PRINT";
80 IF LEN(X$)<>255THENPRINT#1,""
90 GOTO50
```

End of listing.

Listing 2. REATST.BA copies .DO files from other banks.

```
10 'WRITST.BA by Paul Globman
20 ' get input data and determine bank
30 CLS:PRINT:INPUT"Client";CL$
40 IF(ASC(CL$)AND223)=ASC(CL$)THEN60
50 BEEP:PRINT"use CAPS only please"
55 FORI=1TO399:NEXT:RUN
60 IF D$<"I"THENBNK=1:GOTO90
70 IF D$>"Q"THEN BNK=3:GOTO90
80 BNK=2
90 CL1$=CL$+CHR$(13)+CHR$(10)
100 ' open file and locate position
110 F$=CHR$(BNK+48)+"":CLIENT"
120 LOADM"XBASIC":CALL63600,1,VARPTR(F$)
130 P1=PEEK(63603):P2=PEEK(63604)
140 X$="":CALL63600,3,VARPTR(X$)
150 IF PEEK(63605)=17 THEN 180
160 IF X$<CL$ THEN 130
170 ' write data into place
180 POKE63603,P1:POKE63604,P2:POKE63605,
2
190 CALL 63600,4,VARPTR(CL1$)
200 RUN
```

End of listing.

Listing 3. WRITST.BA demonstrates programmer control.

```
*****
*      XBASIC.CO      *
*      by Paul Globman  *
*      Copyright (C) 1988  *
*****
BEEP: EQU 4F45H      ;all errors
BUF: EQU EF70H
CLRPL: EQU 1C1AH
CREATE: EQU 2D7CH
INSERT: EQU 82A8H
PEEK: EQU 9BB1H
POKE: EQU 9BB0H

=====
;= HOOK FROM BASIC WITH FOLLOWING:
;=
;= 10 CALL 63600,CMD
;= :CALL 63600,CMD,VARPTR(X$)
;= :CALL 63600,CMD,BANK
;=
;= CMD FUNCTION VARPTR(X$)
;= -----
;= 0 CLOSE N/A
;= 1 OPEN READ X$ = FILENAME
;= 2 OPEN WRITE X$ = FILENAME
;= 3 GET X$ X$ = DATA
;= 4 PUT X$ X$ = DATA
;= 5 CHAIN BANK (IPL)
;= 6 FILES BANK
;=
=====
org 63572

;
;=====
;The following code will move a block
;of memory in current bank to any
;location in destination bank.
;Destination bank poked to blkmov+2 by
;earlier code.
;BC=len, HL=source adrs, DE=dest adrs
;
```

Continued.

```
blkmov: push b      ;save counter
        mvi b,0     ;dest bank in B
        push d      ;store dest adr
        mov d,m     ;data byte in D
        xthl        ;dest adr in HL
        call poke   ;POKE it!
        pop d        ;restore DE and
        xchg        ;HL registers
        pop b        ;restore BC:len

        inx h        ;increment ptrs
        inx d        ;to do next.
        dcx b        ;decrement len
        mov a,c      ;and test if B
        ora b        ;and C are 0
        jnz blkmov   ;done if BC = 0
        ret

;=====
get_b: di           ;make this peek
        call 9bb1h   ;
        ei           ;
        mov a,d      ;put byte in a
        ret

;=====
entr: jmp cmd

; ptr: dw 00h
; rd: db 00h

; cmd:
        ana a        ;cmd 0
        js close     ;
        cpi 6        ;cmd 6
        js a_file    ;
        cpi 5        ;
        js chain     ;cmd 5
        cpi 4        ;cmd 4
        js write     ;
        cpi 3        ;cmd 3
        js read      ;

        open: mov b,a ;a=1 or a=2
        lda rd      ;test for rao
```

Continued.

```
ana a
jnz opnerr
mov a,b
sta mode+1
sta rd

call getnam

inx h      ;skip #1
inx h
mov e,a    ;filename len
call 5acah ;fix name

lxi h,f746h ;move filename
lxi d,f746h ;to other bank
lxi b,6
call blkmov
jmp both

; opnerr: call beep
; close: lxi h,0
;        shld ptr
;        shld ptr+1
;        ret

; write: sta mode+1
;        lda rd
;        cpi 2
;        jnz beep

;
; push h      ;save varptr()
; mov a,m     ;look at len
; cpi 0       ;is len 0?
; mov l,a     ;
; mvi h,0     ;len in hl
; shld bc+1   ;store len here
; pop h       ;get varptr(x$)
; rz         ;ret if len=0

;
; inx h       ;hl=adrs of ptr
; xchg        ;
; lhl i       ;hl = loc of x$
```

Continued.

BACK ISSUES!

Ever since we bought Portable 100, the most frequently asked question has been, "Do you have any back issues?" Up until now the answer has been, "No." Nevertheless, all this time we have been tracking down a rumor that somewhere in Camden, Maine, was a barn loaded with old issues of Portable 100. Well, we finally ran it to ground and found the barn, the issues, and the barn's owner. To make a long story short, we finally have Portable 100 back issues. The bad news is that we have only a limited number of issues, merely 100 per month published, thirty-one months in total, plus eight of our own (see chart below).

Because of this we are making these collector issues available for \$19.95 each, postage included.

To help you decide which issues you want, we've put together a special, comprehensive 20-page article index covering every issue published from September, 1983 (premier issue) to the July/August 1988 issue. This index is available for only \$5.95, plus \$1.05 shipping and handling. The \$5.95 will be credited against your first back issue order.

Month	83	84	85	86	87	88
January			NOT PUBLISHED		NOT PUBLISHED	OUT
February						
March			OUT			
April		OUT				
May		OUT				
June		OUT				
July		OUT		OUT		
August						
September	Premier Issue			NOT PUBLISHED		NOT YET PUBLISHED
October		OUT				
November						
December						

Total

Not sure which
issue to order?
Get the
article index
and find out
exactly which
issues you need.

- ☐ Please send me the Portable 100 Article Index. I understand that the \$5.95 cost will be credited to my first back issue order.
- ☐ Please send me the back issues I've indicated above, at a cost of \$19.95 each, shipping and handling charges included.

PAYMENT MUST ACCOMPANY ORDER

- ☐ CHECK ENCLOSED ☐ MC ☐ VISA

Total enclosed: _____ CARD# _____ EXP. DATE _____
BE SURE TO INCLUDE ENTIRE CARD NUMBER

NAME _____
ADDRESS _____
CITY _____ STATE _____ ZIP _____

About Your Subscription

Your copy of Portable 100 is sent via second-class mail. You must notify us of your new address when you move. Please allow two months for processing. Sorry, we cannot be responsible for sending another copy when you fail to notify us in time.

Your mailing address also shows an "account number" (first four numbers on the top line) and the subscription expiration date (last six numbers on top line). Please give us this account number and expiration date when renewing or corresponding with us. It will help us help you better and faster.

Simply send your new address and the mailing label from a previous issue to:

Portable 100

145 Grove St. Ext.

P.O. Box 428

Peterborough, NH

03458

or call us at:

(603) 924-9455

Is the screen on your Laptop **SCRATCHED**? Then restore it yourself with...



ULTRALUCENT is a scientifically engineered set of cushioned abrasives that will restore the hard plastic display of any laptop or notebook computer back to "just-out-of-the-box" BRAND-NEW condition. ULTRALUCENT will remove haziness, hairline scratches, minor scratches and deep scratches. "Screen restoration takes just minutes, but the benefits can be seen immediately and are long lasting. And it is so easy to use too!" ... Larry Turner, Photographer, Montreal

ULTRALUCENT is unconditionally & fully guaranteed for 100% satisfaction or your money back! Here's what you get:

- 8 cloth-backed, color coded, re-usable cushioned abrasive pads
- Application foam block
- Anti-Static Finishing Cream for removal of abraded particles
- Wiping towel and complete instructions

All this for only \$19.95 + \$3.00 Shipping & Handling. To order (VISA, MC, CHECK or COD), phone or write:
ULTRASOFT INNOVATIONS INC. 76 Main Street, P.O. Box 247, Champlain, NY 12919 Tel. (514) 487-9293 (9-5 EST)
DEALER INQUIRIES WELCOME! Laptops with soft plastic displays can use our ULTRALUCENT EL (\$14.95 + S. & H.)

Circle 37 on reader service card

```

; shld srce+1 ;store it!
; lhld pntr
; shld wr+1
;
; both: di
; in d8h ;save current
; ani 0ch ;bank
; sta swbck+1
; desp 0 ;and save the
; xchg ;stack pointer!
; shld setsp+1
;
; lxi h,bsw ;move following
; mov d,h ;executable
; mov e,l ;code to dest.
; lxi b,setsp-bsw ;bank (from bsw
; call blkmov ;to setsp).
;=====
; bsw: mvi a,0 ;sw to dest bnk
; out d8h ;and execute!
;
; mov b,a ;only reset sp
; lda swbck+1 ;if we changed
; cmp b ;banks. leave
; jz mode ;sp intact if
; lxi sp,63900 ;no bank change
; call 4c47h ;reset sp
;
; mode: mvi a,0
; cpi 4
; jz wr
; cpi 2
;
; lxi h,f74ch ;add .do to the
; mvi m,'d' ;file name that
; inx h ;was specified.
; mvi m,'o'
; push psw
; call create ;open file.do
; pop psw
; cz 8268h ;find eof (hl)
; jmp swbck
;
; wr: lxi h,00h ;pntr
; bc: lxi b,ffffh ;len in bc
; call insert ;cflag = om?
;
; swbck: mvi a,0 ;sw to src bnk
; out d8h ;and fix stack
;=====
; setsp: lxi sp,0 ;sp reset
; push psw
; lda mode+1
; cpi 4
; jnz exit ;not write cmd
; pop psw
; jc er
; xchg ;dest loc in de
; srce: lxi h,0 ;srce loc in hl
; ;len in bc
;
; call blkmov ;if room, do it
;
; xchg
; push psw
; pop psw
; shld pntr
; ei
; ret

```

```

; er: call beep ;elbe beep om?
; mvi a,12h ;and put 0 at
; sta rd ;rd for err flg
; ei
; ret
;=====
; read: lda rd
; cpi 1
; jnz beep
;
; di
; push h
; lda bsw+1
; mov b,a
;
; mvi c,1 ;count in c
; lxi d,buf ;buffer
; lhld pntr
; dcx h
;
; rl: inx h
; push d
; push b
; call peek
; mov a,d
; pop b
; pop d
;
; cpi 26
; jz eof1 ;end of file
; cpi 13
; jz eor ;end of record
;
; stax d
; inx d
; inr c
; jnz rl
; dcx h
;
; eor: inx h
; inx h
; shld pntr
; mov a,c
; dcr a
; pop h
; mov m,a
; inx h
; xchg
; lxi h,buf
; shli
; ei
; ret
; eof1: mvi a,11h
; sta rd
; jmp eof
;=====
; chain: di
; mov a,1
; call setdst
; lxi h,f4ddh
; push h
; pop d
; lxi b,11
; call blkmov
; call clripl
; lda blkmov+2
; mov c,a
; jmp 9b90h

```

```

;=====
; getnam: mov a,m ;name length
; dcr a
; dcr a
; inx h ;pntr to adrs
; xchg
; lhli ;adrs of name
;
; push psw
; mov a,m ;get bank
; sui 30h ;ascii to bin.
; call setdst
; pop psw ;a=len
; ret
;=====
; setdst: mov b,a ;convert from
; mvi a,1 ;bin(1,2,3)
; chl: add a
; dcr b ;to (0,4,8)
; jnz chl
; ani 0ch
; sta bsw+1 ;dest bank
; sta blkmov+2 ;twice
; ret
;=====
; s_file: mov a,1
; call setdst
; mov b,a
;
; bank: lxi h,f2b5h ;user file #1
; sl: call get_b ;attrib byte
; cpi fffh ;no more files
; rz ;then return
; cpi 240 ;rom filename?
; jz s2 ;don't print it
; ani 80h ;active file?
; jnz file ;yes-show it!
;
; s2: lxi d,11 ;no-get next
; dad d ;file slot and
; jmp sl ;do it again!
;
; file: call leh ;print space
; inx h ;skip attrib
; inx h ;byte and adrs
; c_loop: inx h
; call get_b ;get filename
; rst 4
; skip: mvi a,8 ;any more chrs?
; dcr a
; jz c_done ;no-then done
; sta skip+1 ;adjust counter
; cpi 2 ;time for dot?
; jnz c_loop ;get next chr
; mvi a,'.' ;print the dot
; rst 4
; jmp c_loop
;
; c_done: mvi a,8 ;restore count
; sta skip+1
; inx h ;and do next
; jmp sl ;file slot
;=====

```

End of listing.

The Portable 100 Classifieds

SERVICES

MODEL 100, 102 200 SUPPORT
(Five-Year-Old Club)

Ongoing support for Tandy/Radio Shack Model 100, 102 and 200 owners. International support by mail, voice, and bulletin board. Free on-line weekly newsletter 300/1200 baud, 8-bit, 24hrs/7days, 415-939-1246. Voice recording: 425-937-5039.

For INDEX of public domain library, "how to" bulletins, product reviews, write to: CLUB 100, A Model 100 User Group, 984 Hawthorne Drive, Walnut Creek, CA 94596

Patent Attorney Donavon Faure
(703) 920-2962

\$500.00 Weekly Mailing Circulars in your spare time. Send large self-addressed Stamped Envelope for Details To: H. E. Boulware, Route 1 Box 9, Vichy, Mo, 65580.

SOFTWARE

SPC Programs for 100/102 Disk or Cassette-Quality Resources—840 McKinley, Plymouth, MI 48170

HARDWARE

Model 100 Hardware for sale: Chipmunk Disk Drive with software \$250, 128K RAM Expansion board \$150. Call 216-647-6371

Free Information

For free information on products advertised in this issue of Portable 100, locate the Reader Service number corresponding to the advertisement that interests you. Circle the number on the Reader Service Card at the center of the magazine (or on the wrapper protecting the magazine if you are a subscriber) and drop it into the mail. The literature you've requested will be forwarded to you without any obligation. Please allow 3--5 weeks for delivery.

The Portable 100 Classifieds

HOW TO PLACE A CLASSIFIED AD

Categorize your advertisement (Hardware, Software, Services, Wanted, Etc.) and carefully type your message. We are not responsible for correctly interpreting handwritten advertisements. Phone numbers, street numbers, dimensions, and any abbreviations count as one word. Logos, company or product, are not allowed, neither are display advertisements. Business rates are \$1.60 per word; non-business rates (individuals advertising) are \$1.30 per word. Add up the cost and send the advertisement copy with a check, money order, Visa, or Mastercard number to: Portable 100 Classifieds, 145 Grove St. Ext., P.O. Box 428, Peterborough, NH 03458-0428, c/o Linda Tiernan. Make checks payable to Portable 100 Classifieds. Include your complete return address and phone number (phone number is printed only if it is included in the advertisement itself). Materials due the 1st of the month, two months prior to the magazine cover date (example: materials for the February issue must be received by December 1st.) Advertisements received after the deadline will appear in the next scheduled issue. Payment must accompany order. No refunds for advertisements that miss deadlines, regardless of reason. We reserve the right to change advertisement categories, and to reject, edit, or cancel any advertisement we deem improper. There are no agency discounts for classifieds. For faster service call 603-924-7949.

Ad Category _____ No. of words x (1.60)(1.30) _____ #Issues to run _____

Name _____

Address _____

City _____ State _____ Zip _____

Total Enclosed x (Ad Cost x Issues) _____ Phone Number _____

Visa/Mastercard Number _____ Exp. Date _____

Signature _____

If more space is needed, please use a separate sheet of paper.

ADVERTISER

RS#		Page
88	Rural Engineering	CIII
60	Simons	CIII
29	Software by Steffans	CIII
22	Sound Sight	CII
37	Ultrasoft Innovations	31

** Please contact this advertiser directly

Don't forget
to say
you saw it in
Portable 100!

By Teachers for Teachers Grade Management System

PERSONALIZED ... STAY ORGANIZED

You Set Up To:

12 Grade Values • 9 Classes
50 Students

3½ Disk for PDD
TANDY 100/102 - 200

We also customize 5 basic programs
\$49.00 + \$2.00 (S & H)

Software by Steffens'
1731 William Ct.
Little Chute, WI 54140

Circle 29 on reader service card.

Data Acquisition System For the Model 100/102 and Model 4

Provides engineers, technicians, educators, students and experimenters with a flexible solution to data acquisition and interface needs using the Model 100/102 or Model 4 computer.

M102-DACQ SYS-1 \$185.00

8 digital input lines; 8 digital output lines; 8 bit A/D with 8 channel MUX (ADC0809); single channel 8 bit D/A (DAC0831); 3 channel, 16 bit prog timer (82C53); auto on/off; interfaces to system bus; 3 sq. in. wire wrap area; user manual, schematics, programming instructions/examples. Board, cabling, AC adapter and case (7.5" x 4.33" x 2.22") included.

M102-DACQ-SYS1 \$285.00

DACQ-1 plus a terminal board for easy access to all signals. The terminal board provides: large (12 sq. in.) wire wrap area; diode clamps on analog inputs; block terminals for analog signals; dual row header for digital signals.



Other options available, call or write for more information. Specify (M100-DACQ for Model 100, M4-DACQ for Model 4) interface cables.

Postage Paid, M/C, VISA, money order, check, approved P.O.
Sales Tax: add 4% (VT residents)

Rural Engineering Inc.

Route 14, Box 113F, So. Royalton, VT 05068
Tel. (802) 763-8367

Circle 88 on reader service card.

Not Sure About A Product Or Service You've Seen Here?

Then why not fill out the Reader Service Card right now and send it to us? You'll receive more information directly from the vendor or manufacturer.

TRAVEL CASE PLUS+ IS A PORTABLE MINI WORK STATION



- Computer and disk drive attach to the removable plastic tray
- The Tandy 102 or 200 can remain connected to your disk drive
- Unzip navy blue padded nylon case, adjust typing easel, turn on switch and begin work
- Large front pocket will hold a TTXpress or Diconix 150 Printer
- Dimensions: 12.5" x 14" x 3"



Tray also fits:

- Model 100
- NEC 8201A
- Chipmunk Drive

Case without tray holds:

- NEC Multispeed
- Toshiba 1100+

**SIMONS
PRODUCTS**

10908 GLEN WILDING LANE
BLOOMINGTON, MN 55431
PHONE (612) 881-7221

Travel Case+ \$85.00 • Case w/o tray \$70.00
Return within 15 days if not satisfied.

Check, Money Order, Visa, M/C
(Minnesota residents add 6% sales tax).
Add \$5.00 for shipping/handling in U.S.A.

Circle 60 on reader service card.

The sky's the limit... ...with Portable 100 magazine.

On the road...at home
...in the office...in the
air...the variety of tasks
you can perform with
your Tandy portable is
expanding constantly.

To keep fully informed
and up-to-date about cur-
rent trends, new products,
and new uses for your
laptop, you need **Portable
100** magazine.

From sophisticated input/
output (I/O) calls and
their applications, to sim-
ple disk drives, **Portable
100** magazine covers it all.

Portable 100 gives you
features, news, columns,
and reviews that are
thorough and timely.
And they are written in a
fast-paced, easy-to-read
style, by leading experts
in the computer field.

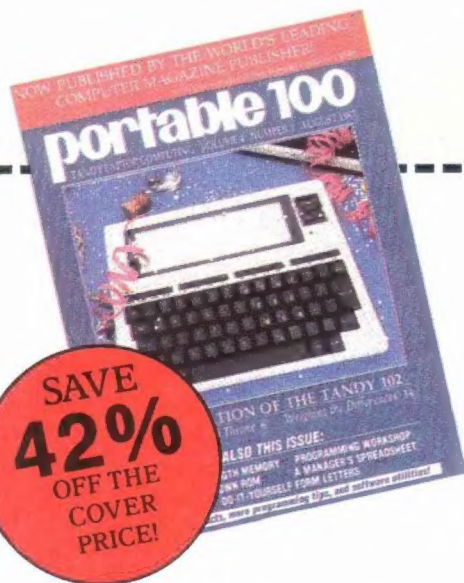
In upcoming issues you
will discover...

- how new peripherals
can make you more
productive.
- more efficient ways
to communicate with
your desktop.
- where to buy low-cost
public-domain
software.
- how your business
can be more profitable
with your portable at
your fingertips.
- where to find the best
bulletin boards and
information services.
- and much more!

Don't miss even a single
valuable monthly issue.
Fill out the coupon below,
or to charge it to your
credit card, call

1-603-924-7949

Send no money now! We
will gladly bill you.



☐ **YES!** I want to be more productive with my Tandy portable... and save 42% off the newsstand price. Send me a year's subscription (monthly except for a combined July/August issue) of Portable 100 for us\$24.97.

☐ Payment enclosed ☐ Bill me
Make checks payable to **Portable 100**

Name _____

Address _____

City _____ State _____ Zip _____

Canada and Mexico, us\$29.97, Foreign surface,
us\$44.97, Foreign Air Mail add us\$50. All foreign,
except Canada, US Funds drawn on US bank.
Please allow six weeks for delivery. LTP

MONEY BACK GUARANTEE:

If you are not completely
satisfied with **Portable 100**,
you may cancel your sub-
scription and receive a full
refund. Please allow 6-8
weeks for delivery of your
first issue.